WL-TR-97-4014

3D BOUNDARY ELEMENT ANALYSIS FOR COMPOSITE JOINTS WITH DISCRETE DAMAGE

Part 1: Report

Paul A. Wawrzynek
Bruce J. Carter
Leonard J. Gray
Anthony R Ingraffea

Fracture Analysis Consultants, Inc.
121 Eastern Heights Drive
Ithaca NY 14850-6345

NOVEMBER 1996

FINAL REPORT FOR 15 APR 96 -- 15 NOV 1996

Approved for public release; distribution unlimited

MATERIALS DIRECTORATE
WRIGHT LABORATORY
AIR FORCE MATERIAL COMMAND
WRIGHT-PATTERSON AFB OH 45433-7734

NOTICE

WHEN GOVERNMENT DRAWINGS, SPECIFICATIONS, OR OTHER DATA ARE USED FOR ANY PURPOSE OTHER THAN IN CONNECTION WITH A DEFINITE GOVERNMENT-RELATED PROCUREMENT, THE UNITED STATES GOVERNMENT INCURS NO RESPONSIBILITY OR ANY OBLIGATION WHATSOEVER. THE FACT THAT THE GOVERNMENT MAY HAVE FORMULATED OR IN ANY WAY SUPPLIED THE SAID DRAWINGS, SPECIFICATIONS, OR OTHER DATA, IS NOT TO BE REGARDED BY IMPLICATION, OR OTHERWISE IN ANY MANNER CONSTRUED, AS LICENSING THE HOLDER, OR ANY OTHER PERSON OR CORPORATION; OR AS CONVEYING ANY RIGHTS OR PERMISSION TO MANUFACTURE, USE, OR SELL ANY PATENTED INVENTION THAT MAY IN ANY WAY BE RELATED THERETO.

THIS REPORT IS RELEASABLE TO THE NATIONAL TECHNICAL I NFORMATION SERVICE (NTIS). AT NTIS, IT WILL BE AVAILABLE TO THE GENERAL PUBLIC, INCLUDING FOREIGN NATIONS.

THIS TECHNICAL REPORT HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION.

Richard B. Hall 2/3/97
RICHARD B. HALL, Program Manager
Mechanics & Surface Interactions Branch
Nonmetallic Materials Division

Paul J Hesse 3 JAN 97
PAUL J. HESSE, MAJ, USAF, Chief
Mechanics & Surface Interactions Branch
Nonmetallic Materials Division

Charles E. Browning
CHARLES E. BROWNING, Chief
Nonmetallic Materials Division
Materials Directorate

IF YOUR ADDRESS HAS CHANGED, IF YOU WISH TO BE REMOVED FROM OUR MAILING LIST, OR IF THE ADDRESSEE IS NO LONGER EMPLOYED BY YOUR ORGANIZATION, PLEASE NOTIFY WL/MLBM, WRIGHT-PATTERSON AFB OH 45433-7750 TO HELP MAINTAIN A CURRENT MAILING LIST.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	15 NOV 96	SBIR P 1 FINAL 15 APR 96 -- 15 NOV 96

4. TITLE AND SUBTITLE

3D Boundary Element Analysis for Composite Joints with discrete Damage, Part 1; Report

5. FUNDING NUMBERS

C F33615-96-C-5070
PE 65502F
PR 3005
TA 05
WU S3

6. AUTHOR(S)

Paul A. Wawrzynek Leonard J. Gray
Bruce J. Carter Anthony R. Ingraffea

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Fracture Analysis Consultants, Inc.
121 Eastern Heights Dr.
Ithaca, NY 14850-6345

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Materials Directorate
Wright Laboratory
Air Force Materiel Command
Wright-Patterson AFB OH 45433-7734

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

WL-TR-97-4014

11. SUPPLEMENTARY NOTES

THIS IS A SMALL BUSINESS INNOVATION RESEARCH (SBIR) PHASE i REPORT.
Part II - Software (WL-TR-97-4015)

12a. DISTRIBUTION AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

WL/MLBM solicited Phase I SBIR proposals to develop a capabiliity for 3D boundary element analysis for composite joints with discrete damage. Fracture Analysis Consultants, Inc. with its extensive experience with boundary element technology and existing boundary element software, responded to the solicitation, and was granted a six month contract to demonstrate the feasibility of such a system. There were three tasks in the workstatement. All three have been completed successfully. These include: completion of a, 2D, symmetric-Galerkin code, validation of the code on a test problem, and demonstration of capabilities for 3D contact problems. In addition considerable resources were spent on three additional tasks: addition of fracture mechanics capabilities to the 2D code, development of an efficient general routine for computing 3D orthotropic Green's functions, and implementation of a fully 3D symmetric-Galerkin code (currently in the testing and debugging phase).

14. SUBJECT TERMS

Boundary Elements, Composite Joints, Damage, Fracture Mechanizes, Contact Stress

15. NUMBER OF PAGES 50

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

Fracture Analysis Consultants, Inc.

121 Eastern Heights Dr. Ithaca, New York 14850 (607) 257-4970

SBIR Topic AF96-150

3D Boundary Element Analysis for Composite Joints with Discrete Damage

Contract #F33615-96-C-5070

Phase I Final Report

Paul A. Wawrzynek, Ph.D.,
Bruce J. Carter, Ph.D.,
Leonard J. Gray, Ph.D. (Consultant), and
Anthony R. Ingraffea Ph.D. (Consultant)

November 15, 1996

Contents:

Summary Report

WL/MLBM solicited Phase I SBIR proposals to develop a capability for 3D boundary element analysis for composite joints with discrete damage. Fracture Analysis Consultants, Inc. (FAC), with its extensive experience with boundary element technology and existing boundary element software, responded to the solicitation, and was granted a six month contract to demonstrate the feasibility of such a system.

Our overall goal for was to demonstrate expertise in BEM methods appropriate to the detailed stress analysis of arbitrary bolted or bonded composite joints containing cracks and contact surfaces. There were three tasks in the Phase I workstatement. All three have been completed:

Task 1. We augmented the Galerkin-based, 2D, orthotropic BEM program developed for the original program solicitation to include *hypersingular* BEM traction equations so that a symmetric coefficient matrix could be formulated. We believe that this orthotropic symmetric-Galerkin code is unique.

Task 2. Using the code developed in Task 1, we solved an example laminate problem [Pagano 1978], to show that the symmetric-Galerkin (SG) approach can yield accurate interlaminar stress predictions near free surfaces.

Task 3. We demonstrated a capability to perform 3D contact analysis using our existing 3D BEM program. The approach is a penalty function technique derived from a current capabilities for fluid driven fracture, and can also be used for the proposed Phase II 3D simulation system.

In addition to these tasks, considerable resources were spent on three additional tasks that further our goal of demonstrating expertise, and are preparatory for a potential Phase II effort.

Task 4. We have added fracture analysis capabilities to the 2D SG code of Task 1. We believe that this is the first SG fracture formulation for elasto-statics, and certainly the first for orthotropic materials.

Task 5. We have developed software to evaluate the 3D anisotropic Green's function. This is a generalization of the work of Sales and Gray [1996]. In this approach residue calculus is used to transform the Green's function to a ratio of polynomials. A symbolic algebra program is used to generate polynomial coefficients. Newton's method is used to find the roots of the polynomial. Our implementation can handle multiple orthotropic materials; no precomputations are required.

Task 6. We have developed a fully 3D Galerkin BEM program. Coding for this program has been completed, we are currently in the testing and debugging phase. This code includes linear and quadratic order elements, and has been interfaced with our FRANC3D program that we use to generate models, meshes, and boundary conditions.

In summary, we have successfully completed the work tasks specified in our proposal. In addition, we have gone substantially beyond the original Phase I objectives towards creating a capability for state-of-the-art 3D boundary element analysis of composite joints with discrete damage.

1. Introduction and Background

WL/MLBM is developing improved capability for detailed stress analysis of composite bolted and bonded joints. Such joints are crucial to the success of composite structures because they are sites for initiation of a variety of damage types, and contribute substantially to weight and to design and manufacturing resources.

Geometry changes, ply drops, and free-edges common to such joints create stress concentrations [Pagano, 1978] and theoretical stress singularities [Wang & Crossman, 1977] which render simple lamination theory inapplicable. A mathematical model whose characteristic dimension is lamina thickness is required to approach these problems [Chamis, 1989]. Although there have been some successful solution techniques based on more rigorous analytical approaches at this scale, these are restricted in applicability because they lack the geometrical and boundary condition generality of numerical methods such as the finite element method (FEM).

However, there has been a historical difficulty in using the FEM for detailed stress analysis of laminates. This is because of the intrinsic difficulty with the FEM of simultaneously satisfying displacement and traction compatibility across lamina interfaces while also providing the strain discontinuity as required by the orthotropic constitutive laws of the lamina materials [Bogdanovich, 1992].

The boundary element method (BEM) appears to be an attractive alternative numerical approach for this application because it automatically satisfies displacement and traction compatibility across lamina interfaces while naturally reproducing the required strain discontinuity.

WL/MLBM solicited SBIR proposals for the development of a 3D BEM capability both to compare with in-house variational methods of solution, and to improve efficiency and applicability. Fracture Analysis Consultants, Inc. (FAC), with its extensive experience with boundary element technology and existing boundary element software, responded to the solicitation, and was granted a six month Phase I contract to demonstrate the feasibility of such a system. This is the final report for this Phase I effort.

1.1 Phase I Technical Objectives

The overall goal of Phase I is to demonstrate expertise in BEM methods appropriate to the detailed stress analysis of arbitrary bolted or bonded composite joints containing cracks and contact surfaces.

Technical objectives outlined in the proposal were:

Objective 1: Modification of a new Galerkin 2D BEM code to generalized plane strain and to symmetric formulation so that we can solve the specified test problem referenced in the solicitation [Pagano, 1978]. This will demonstrate the feasibility of the symmetric Galerkin boundary element formulation.

Objective 2: Produce a fully 3D solution to the test problem. This would then enable comparisons for accuracy and efficiency to the predictions produced from Objective 1.

Objective 3: Modify FRANC3D (our current boundary element analysis system) to enable solution of contact stress problems. This will allow us to analyze arbitrary bolted or bonded composite joints containing contact surfaces.

Objectives 1 and 3 have been fully realized. Objective 2 has been only partially fulfilled. Our original intention for Objective 2 was to modify our existing, collocation-based finite element code to include anisotropic material capabilities. However, during the course of the project we decided that it would be more useful in the long run to develop a new 3D Galerkin formulation boundary element code (see the next three sections for definitions of these terms). We have developed this new capability, but because this effort was more than that of the original obective 2, we have not yet been able to perform a fully 3D solution of the test problem.

A detailed description of the results of all tasks associated with these objectives is provided in Section 2. As noted there, even though objective 2 had been only partialy fullfilled, all tasks have been completed, along with a number of additional tasks not in the original workstatement. First, however, we provide a brief background on boundary element formulations and computational fracture analysis therewith. These sections proved a technical context for the remainder of the report.

1.2 Boundary Element Formulations

Conventionally, the boundary element is formulated as a collocation procedure. That is, the governing equation

$$u_{kk}(P)+\int T_{kk\,jj}(P,Q)\,u_{jj}(Q)\,dQ=\int U_{kk\,jj}\tau_{jj}(Q)\,dQ \tag{1-1}$$

is forced to be satisfied at some number of nodal points, on the boundary. The equation is not necessarily satisfied away from these points. While P is a point, the integration is performed over a surface element Q. This is shown pictorially in Figure 1-1a. The integration is relatively straightforward, except for the cases where the P node is adjacent to the Q element. As the traction and displacement kernels, U and T, contain factors of $1/r$ and $(q_k$-$p_k)/r^3$, respectively (q_k and p_k being the Cartesian coordinates of the source and field points), these integrations become singular, and special care must be taken in their evaluation. Techniques for this are well known, and can be found in any number of standard BEM textbooks [e.g., Brebbia and Dominguez, 1989].

While the collocation technique forces the governing equations to be satisfied at a finite number of points, the finite element technique has been very successful using the Galerkin form of the method of weighted residuals. In this approach, the governing equations are not satisfied at individual points, but rather in an average sense over an element's domain. This approach can be applied to boundary elements [Maier and Polizzotto 1987] but requires two integrations, one each over the P and Q elements.

Symbolically, the Galerkin governing equations are

$$\begin{aligned}&\int \psi_k(P)u_{kk}(P)\,dP+\int \psi_k(P)\int T_{kk\,jj}(P,Q)\,u_{jj}(Q)\,dQdP\\&=\int \psi_k(P)\int U_{kk\,jj}\tau_{jj}(Q)\,dQdP\end{aligned} \tag{1-2}$$

where ψ are the shape functions of the P element. This is shown in Figure 1-1b.

There are at least five advantages of the Galerkin formulation:

1. The technique to yields much more accurate internal stresses and displacements close to the boundaries than does the collocation approach. Such stresses and displacements are notoriously bad with collocation giving rise to the so-called "boundary effect". In essence, since the governing equations are satisfied only at nodal points, evaluating stresses or displacements at points near the boundary can give rise to significant errors. The recommendation is to not trust

values evaluated closer to the boundary than the local characteristic nodal spacing.

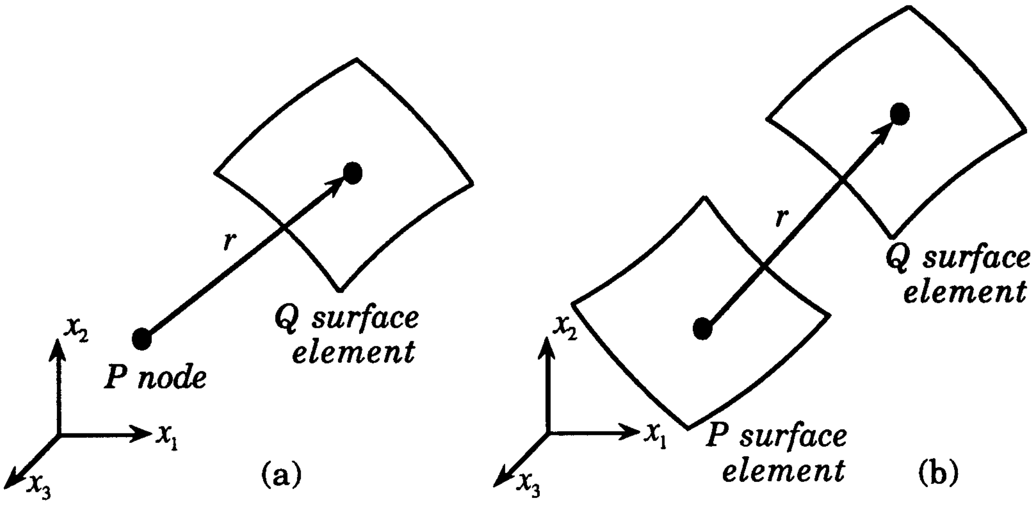


Figure 1-1. a) normal collocation boundary element integration, b) Galerkin boundary element integration.

Such restrictions on evaluating stresses and displacements are quite severe in the present case where we are concerned with inter-laminar stresses, and stress fields in near crack-tip regions. Therefore, the Galerkin formulation seems superior for these cases.

2. The technique gives one the ability to rigorously treat hypersingular equations using standard continuous elements, collocation requiring a differentiable approximation (this is discussed further below).

3. A more accurate treatment of corners, especially for corners having displacement boundary conditions and corners having no boundary conditions. This is important for bi-material interface problems and potentially for coupling BEM domains to FEM domains.

4. In conjunction with the hypersingular boundary element equations, a symmetric coefficient matrix can be formulated, giving rise to significant decreases in necessary storage and processing time.

5. A symmetric formulation give the possibility to couple boundary elements with finite elements and still maintain the symmetric FEM coefficient. matrix.

1.3 Boundary Elements and Fracture Mechanics.

A naive application of boundary element techniques to fracture mechanics fails. This is illustrated in Figure 1-2a. Nodes i and j, while being on distinct surfaces, are geometrically coincident. This means that the boundary element equations for these two points will be identical, leading to a rank deficient coefficient matrix.

The remedy for this situation has been to break the object into two separate domains by way of a fictitious boundary [Rizzo and Shippy 1968, Cruse 1969, Blanford, Ingraffea, and Liggett 1981]. Separate boundary element coefficient matrices are formulated for the two domains, and equilibrium and compatibility is enforced along the boundary. This is illustrated in Figure 1-2b.

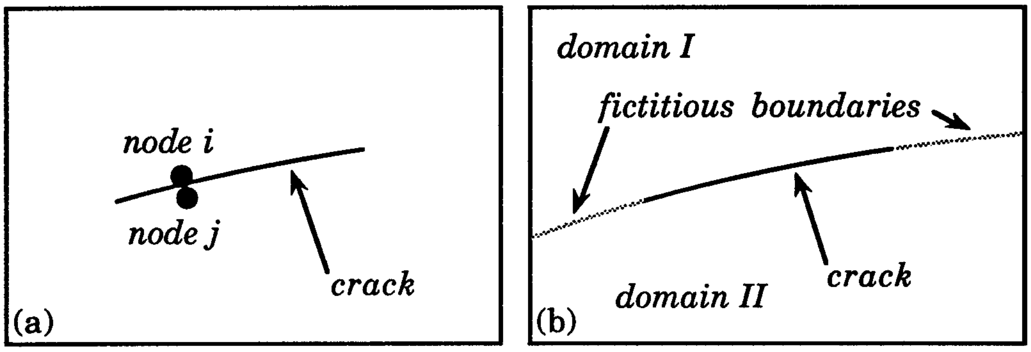


Figure 1-2. a) a naive application of boundary elements to a fracture mechanics problem, b) the multi-domain approach.

The multi-domain approach has been shown to be unsatisfactory for at least two reasons. First, while for two dimensional problems defining a suitable fictitious boundary is usually straightforward, doing the same for an arbitrary three-dimensional configuration can be quite difficult.

Second, this approach adds additional surfaces in the crack tip region. As mentioned above, while the governing equations are satisfied exactly in the interior of the domain, they are only approximately satisfied near the boundaries. The multi-domain approach therefore places additional approximations in the crack-tip region, where stress gradients are expected to be the highest, and where usually, the most accurate results are desired.

Gray and his associates have taken a different approach to the crack problem, where the redundant crack face equations are replaced with equations for the known (usually zero) stress conditions on the crack faces [Gray 1989, Gray, Martha, and Ingraffea 1990, Lutz, Gray, and Ingraffea 1991, Martha *et. al.* 1992, Gray 1993].

In his approach, Eq. (1-1) is differentiated with respect to P to yield the following equation for the tractions,

$$\tau_{kk}(P)+\int S_{kkjj}(P,Q)u_{jj}(Q)dQ=\int W_{kkjj}(P,Q)\tau_{jj}(Q)dQ \qquad (1\text{-}3)$$

which can be substituted for the redundant crack-face equations.

The S and W kernels of Eq. (1-3) contain first and second derivatives of the Green's function, and thus lead to so-called *hypersingular* equations. Gray has shown how these can be evaluated accurately by decomposing the integrals into the non-singular parts, integrated numerically, and simple singular parts, evaluated analytically.

The field of hypersingular boundary integral equations has opened possibilities for new solutions to various problems by the BEM [see, for example, the review article by Krishnasamy, Rizzo, and Rudolphi 1992]. There are a number of hypersingular approaches, and each approach has been developed independently by several groups: e.g. Bonnet and Bui [1993], Crouch and Selcuk [1992], Cruse [1988], Chang and Mear [1995], Gray, Martha, and Ingraffea [1990], Guimarães and Telles [1994], Hong and Chen [1988], and Paulino [1995]. Although successful, the common problem for these methods is that, when employed in conjunction with a *collocation* approximation, a C^1 continuous smoothness constraint is necessarily imposed on the boundary in the neighborhood of the integration point. This condition adds considerable complexity and/or cost to fracture analyses. Historically, we have satisfied this condition by moving the collocation points from the element nodal points to points on the interior of the elements, with the corresponding cost of significantly larger system of equations.

However, if Eq. (1-3) is evaluated with a Galerkin formulation, the resulting equation,

$$\begin{aligned}&\int \psi_k(P)\tau_{kk}(P)dP + \int \psi_k(P)\int S_{kkjj}(P,Q)u_{jj}(Q)dQdP \\ &= \int \psi_k(P)\int W_{kkjj}(P,Q)\tau_{jj}(Q)dQdP\end{aligned} \tag{1-4}$$

can be evaluated with only C^0 continuity between elements.

1.4 A Symmetric Galerkin Formulation

The key to obtaining a symmetric coefficient matrix are the properties of the kernel functions [Sirtori *et. al.* 1992],

$$\begin{aligned}U(P,Q) &= U(Q,P)\\ T(P,Q) &= -T(Q,P)\\ W(P,Q) &= W(Q,P)\\ S(P,Q) &= -S(Q,P)\end{aligned} \tag{1-5}$$

combined with a Galerkin approximation, and appropriate selection of equations for each degree of freedom. The displacement equation (1-2) is used for all degrees of freedom where displacements are specified, and the traction equation (1-4) is used for the remaining degrees of freedom where tractions are specified. After discretization, the resulting set of equations can be written in the following block matrix notation,

$$\begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} \begin{Bmatrix} u_k \\ u_u \end{Bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{Bmatrix} \tau_u \\ \tau_k \end{Bmatrix} \tag{1-6}$$

where the *k* and *u* subscripts represent, respectively, the known and unknown components of the displacement and traction vectors.

Rearranging Eq. 1-6 into the form *[A]{x} = {b}*, and multiplying the second row (hypersingular equations) by -1, one obtains,

$$\begin{bmatrix} -G_{11} & H_{12} \\ G_{21} & H_{22} \end{bmatrix} \begin{Bmatrix} \tau_u \\ u_u \end{Bmatrix} = \begin{Bmatrix} -H_{11}u_k + G_{12}\tau_k \\ H_{21}u_k - G_{22}\tau_k \end{Bmatrix} \tag{1-7}$$

The symmetry of the coefficient matrix, $G_{11} = G_{11}^T$, $H_{22} = H_{22}^T$, and $H_{12} = G_{21}^T$, now follows from the properties of the kernel functions, Eq. (1-5).

2. Workstatement Tasks

There were three tasks in the proposed work statement of this project. The first two were aimed at demonstrating that a symmetric Galerkin boundary element formulation would work, and are continuations of the analysis included in our Phase I proposal. The third task was looking forward to potential Phase II work, and was to investigate the possibility of incorporating contact analysis into a 3D boundary element analysis. All three tasks have been completed.

2.1 Task I - Creation of a 2D Symmetric Galerkin Code

The program solicitation for this project requested that all proposals contain a plane strain, thermal-mechanical analysis of a [0°/90°]s laminate subjected to a uniform temperature change of -100°F. At the time of the solicitation, we had no two-dimensional BEM capabilities (while we did have three-dimensional capabilities). Therefore, we developed a new two-dimensional code for the purpose of responding to the solicitation.

This new code was a Galerkin formulation, but not symmetric. That is, it used the displacement equation (1-2) but did not incorporate the traction equation (1-4). We proposed, as Task I, to augment our 2D code to incorporate a symmetric formulation, and to include capabilities for generalized plane strain.

A symmetric formulation has been implemented. There is essentially no difference in computed results between the symmetric and unsymmetric results. The time to formulate elements is essentially the same for the two programs, while the solution time for the symmetric case is approximately one half that of the unsymmetric version.

The generalized plane strain capability is a near trivial extension of the way we implemented thermo-mechanical loading. Assume that the plane of analysis is the *y-z* plane. Given the full three dimensional constitutive relationships,

$$\begin{Bmatrix}\sigma_x\\ \sigma_y\\ \sigma_z\\ \tau_{xy}\\ \tau_{xz}\\ \tau_{yz}\end{Bmatrix}=\begin{bmatrix}C_{11} & C_{12} & C_{13} & 0 & 0 & 0\\ C_{12} & C_{22} & C_{23} & 0 & 0 & 0\\ C_{13} & C_{23} & C_{33} & 0 & 0 & 0\\ 0 & 0 & 0 & C_{44} & 0 & 0\\ 0 & 0 & 0 & 0 & C_{55} & 0\\ 0 & 0 & 0 & 0 & 0 & C_{66}\end{bmatrix}\begin{Bmatrix}\varepsilon_x\\ \varepsilon_y\\ \varepsilon_z\\ \gamma_{xy}\\ \gamma_{xz}\\ \gamma_{yz}\end{Bmatrix} \qquad (2\text{-}1)$$

and the initial strain vector due to a constant temperature change,

$$\begin{bmatrix}\alpha_x\Delta T & \alpha_y\Delta T & \alpha_z\Delta T & 0 & 0 & 0\end{bmatrix}^T \qquad (2\text{-}2)$$

the expression for the in-plane stresses is

$$\begin{Bmatrix}\sigma_y\\ \sigma_y\\ \tau_{yz}\end{Bmatrix}=\begin{bmatrix}C_{22} & C_{23} & 0\\ C_{23} & C_{33} & 0\\ 0 & 0 & C_{66}\end{bmatrix}\begin{Bmatrix}\varepsilon_y\\ \varepsilon_z\\ \gamma_{yz}\end{Bmatrix}-\begin{bmatrix}C_{22} & C_{23} & 0\\ C_{23} & C_{33} & 0\\ 0 & 0 & C_{66}\end{bmatrix}\begin{Bmatrix}\alpha_y\\ \alpha_z\\ 0\end{Bmatrix}\Delta T-\begin{Bmatrix}C_{12}\\ C_{13}\\ 0\end{Bmatrix}\alpha_x\Delta T\ . \qquad (2\text{-}3)$$

In our code, the body stress terms

$$\begin{Bmatrix}\overline{\sigma}_y\\ \overline{\sigma}_z\end{Bmatrix}=-\begin{bmatrix}C_{22} & C_{23}\\ C_{23} & C_{33}\end{bmatrix}\begin{Bmatrix}\alpha_y\\ \alpha_z\end{Bmatrix}\Delta T-\begin{Bmatrix}C_{12}\\ C_{13}\end{Bmatrix}\alpha_x\Delta T \qquad (2\text{-}4)$$

are added to account for the thermal strains. A similar approach can be taken for generalized plane strain. However, in this case the body stress terms are

$$\begin{Bmatrix}\overline{\sigma}_y\\ \overline{\sigma}_z\end{Bmatrix}=\begin{Bmatrix}C_{12}\\ C_{13}\end{Bmatrix}\varepsilon_x, \qquad (2\text{-}5)$$

which is numerically the same expression as 2-4 with $\alpha_y = \alpha_z = 0$, and $\varepsilon_x = \alpha_x\Delta T$.

2.2 Task II - Solution of a Test Problem

To demonstrate the capabilities added in Task I, a problem originally examined by Pagano [1978] was reanalyzed as Task II. The problem is the [0°/90°]s laminate subjected to a uniform x strain, as shown in Figure 2-1.

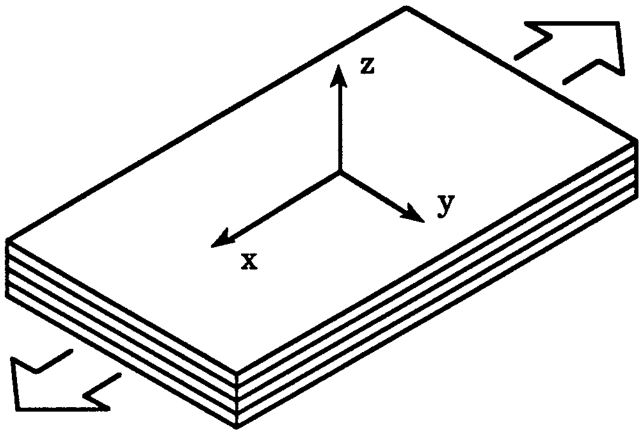
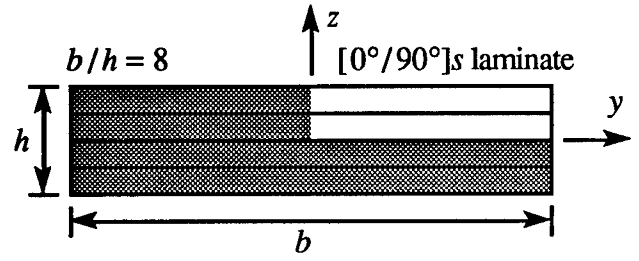


Figure 2-1. A [0°/90°]s laminate subjected to uniform strain in the x - direction.

The laminate is modeled using generalized plane strain, and only one quarter of the cross-section was modeled, as shown in Figure 2-2.



$$E_{11} = 20\times10^{6}\,\text{psi}$$
$$E_{22} = E_{33} = 2.1\times10^{6}\,\text{psi}$$
$$G_{12} = G_{13} = G_{23} = 0.85\times10^{6}\,\text{psi}$$
$$\nu_{12} = \nu_{13} = \nu_{23} = 0.21$$

Figure 2-2. The quarter symmetry model and material properties for the test problem.

The computed inter-lamina stress distributions along the 0°/90° interface are shown in Figures 2-3 and 2-4 (these are directly comparable to Figures 7 and 8 of the Pagano paper).

As can be seen in Figure 2-3, there is very good correlation between our computed normal stress distribution and that reported by Pagano (note, the Pagano results plotted here were determined graphically from the figures in the paper and thus are only approximate).

The comparison for the shear stresses is not quite as good. As shown in Figure 2-4, the present analysis predicts higher peak shear stresses than the reference analysis. We have not investigated the effect of mesh refinement.

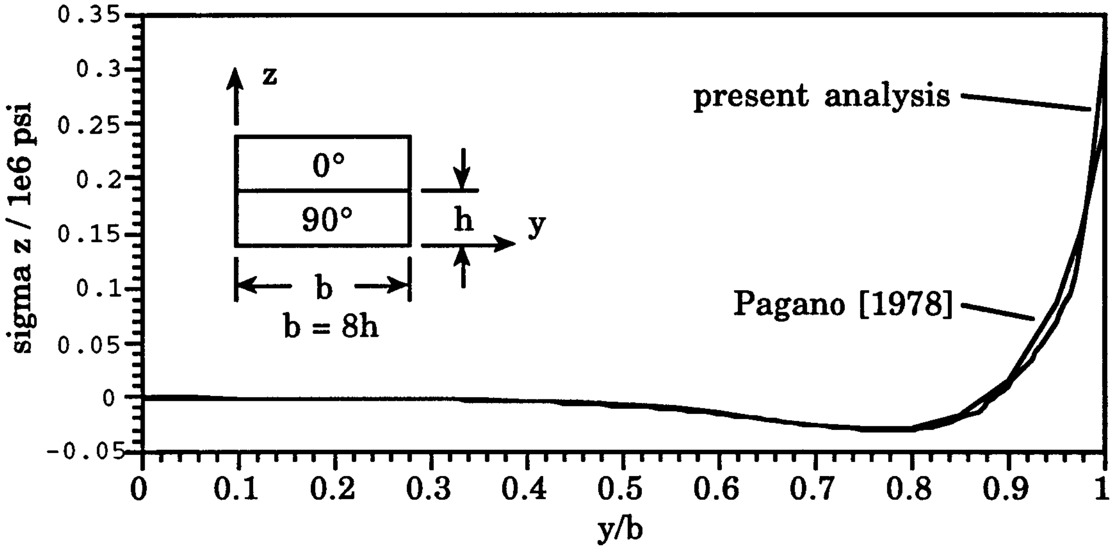


Figure 2-3. Comparison of the normal inter-lamina stresses for the present analysis and that of Pagano [1978].

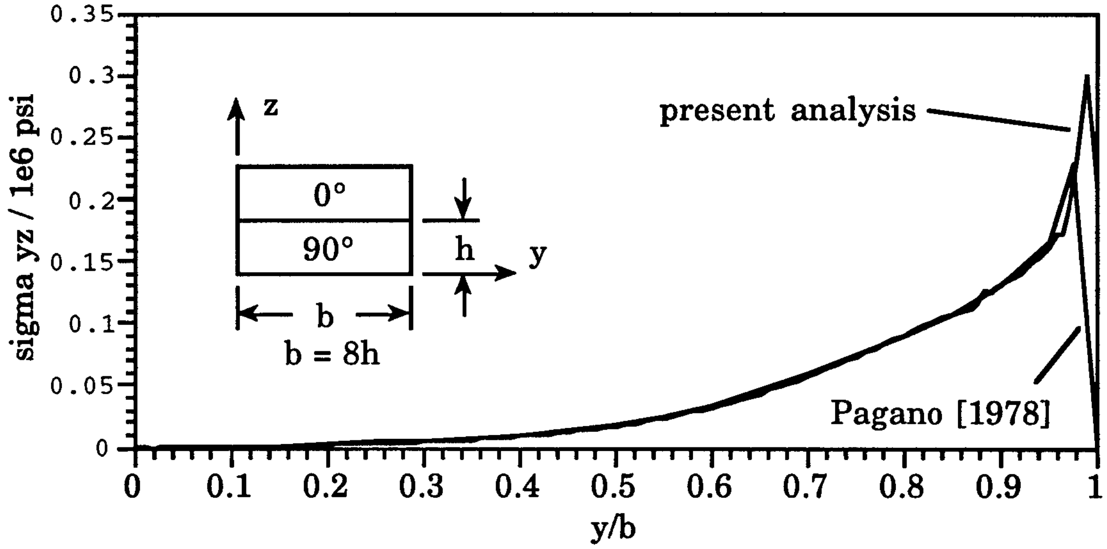


Figure 2-4. Comparison of the shear inter-lamina stresses for the present analysis and that of Pagano [1978].

2.3 Task III - 3D BEM Contact Analyses

An ultimate objective of this project is to develop a detailed analysis capability for mechanically fastened joints in composite materials. In such a joint, bearing stresses from bolts play a major role in the load transfer through the joint, and must be modeled properly in a detailed local stress analysis. Such an analysis must include elastic deformation of the bolt and evaluation of contact zones. In the Phase I effort, we have demonstrated this capability as a straightforward technology transfer from our current coupled fluid/structure interaction [Sousa *et al*, 1993] and non-linear cohesive fracture capabilities [Bittencourt and Ingraffea, 1995].

An important application of our current boundary element program is the analysis of fluid driven fracture (hydraulic fracture). In this case, cracking is driven by pressurized, non-Newtonian fluids inside the crack. This is a highly nonlinear, coupled fluid/structure interaction problem. The nonlinearity arises because the flow in the crack is a cubic function of the crack opening. The solid, however, remains linear and elastic in behavior.

We solve this problem by using the boundary element program to generate a flexibility matrix for the crack surface, which relates the displacement at any point of the crack surface to a load at any other point on the surface. In essence, this is using the boundary element program to create a super element. Once the flexibility matrix has been generated, we use a nonlinear solution algorithm for solving the fluid-structure interaction equations. The important point here is that, at each iteration in the solution, we only need to perform a matrix multiply to incorporate the effects of the solid in the solution.

We have adapted our fluid/structure interaction capabilities, incorporating a penalty function approach, for solving contact problems. The general contact problem has been modeled in 3D using FRANC3D [Wawrzynek 1991] and our existing BEM program, BES. The contact surface is modeled as two distinct surfaces that share the same geometric description. These surfaces are meshed with the same surface mesh. Multiple solutions are obtained during the solution process; one corresponds to the far field applied loading, while the others correspond to unit tractions applied to each node pair on the contact surface. This provides a set of influence coefficients that can be used to determine the equilibrium displacements and tractions due to unit tractions applied to the contact surfaces.

An iterative scheme is used to determine the equilibrium tractions on the contact surface. The algorithm is summarized below:

1. Assume that all tractions are zero on the contact surface, and the amount of overlap due to the far field loading is determined;

2. At the node where maximum overlap occurs, a traction is applied. The traction is computed by comparing the displacements between the node pair on the contact surfaces for the cases of applied far field loading and a unit traction on the node pair (unit tractions are applied in the opposite sense to the two nodes);

3. Based on the ratio of the displacements and a numerical damping (relaxation) factor, a traction value is computed and applied to the node pair;

4. The solution for the far field loading is then superimposed with the solution for the applied traction on the contact surface assuming linear elasticity;

5. The total displacement and traction field is then recomputed;

6. A search is made again for the node pair with the maximum overlap, and the process is repeated, summing up all solutions for all non-zero tractions and computing new displacements and tractions at each iteration.

As an example, a model similar to that analyzed by Yogeswaren and Reddy [1988] was considered. It should, however, be emphasized that while Yogeswaren and Reddy performed plane strain calculations, the present analysis is a fully three-dimensional treatment. They were modeling experiments performed by Joh [1986]. The model consists of an isotropic aluminum plate with a hole and pin. The pin is fixed and tension is applied to the plate. The model, material parameters and loading are shown in Figure 2-5. Appropriate restraints are applied to prevent rigid body motion. The plate and pin are 0.06" thick. Friction on the contact surface is not modeled. Yogeswaren and Reddy modeled a clearance fit pin with 0.005" between the pin and the hole, and they included friction on the contact surfaces; both a static and dynamic coefficient of friction were used. In the present analysis, a slip-fit was assumed, and fictional effects were ignored.

For the first case analyzed, the pin has the same material properties as the plate. The displacements due to far field loading alone are shown in Figure 2-6a and the actual displacements after accounting for contact are shown in Figure 2-6b. Normal tractions on the contact surface are shown in Figure 2-7. Zero degrees corresponds to position A in Figure 2-5. The minimum traction (maximum compression) exists at about 65 degrees. This is also indicated by the color contour plot of minimum principal stress (Figure 2-8).

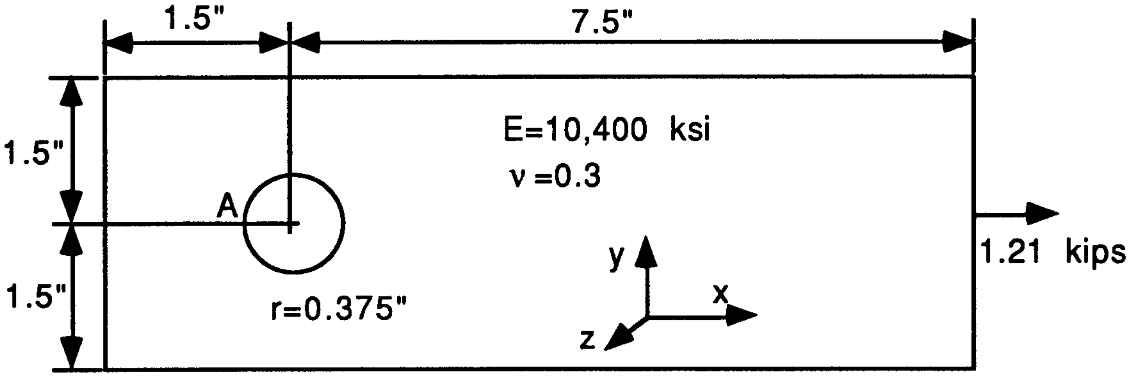


Figure 2-5. Model, material parameters and loading.

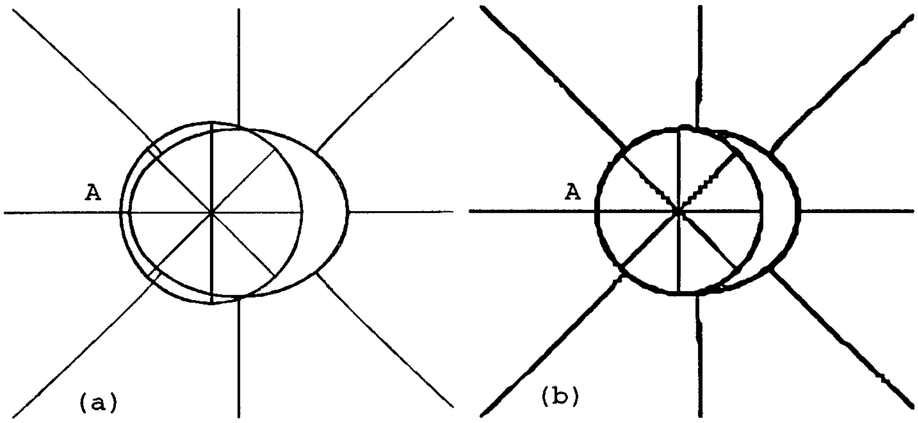


Figure 2-6. (a) displaced shape due to far field loading only, and (b) displaced shaped after accounting for contact. Magnification factor is 100.

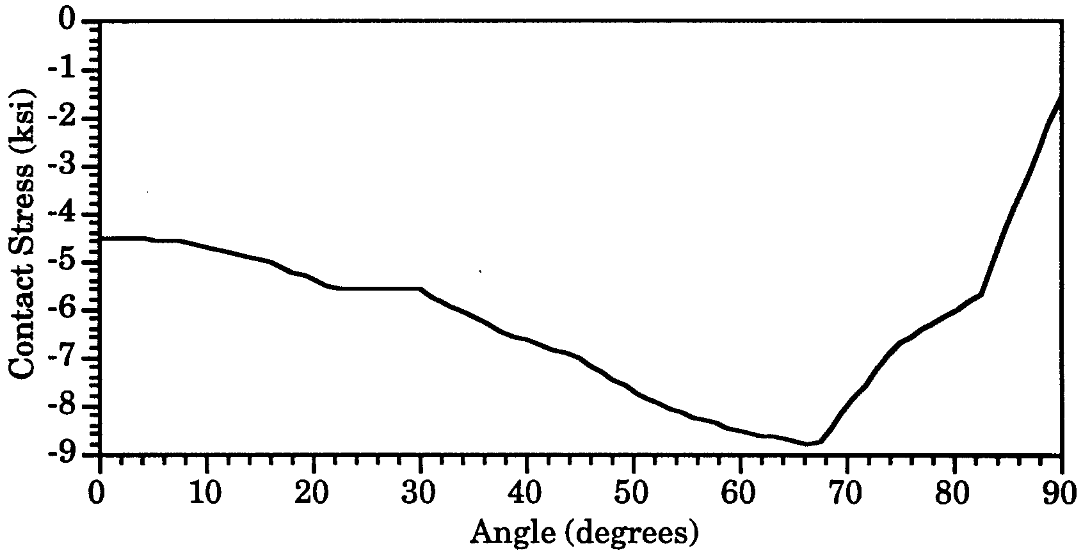


Figure 2-7 Contact normal traction results for the aluminum pin.

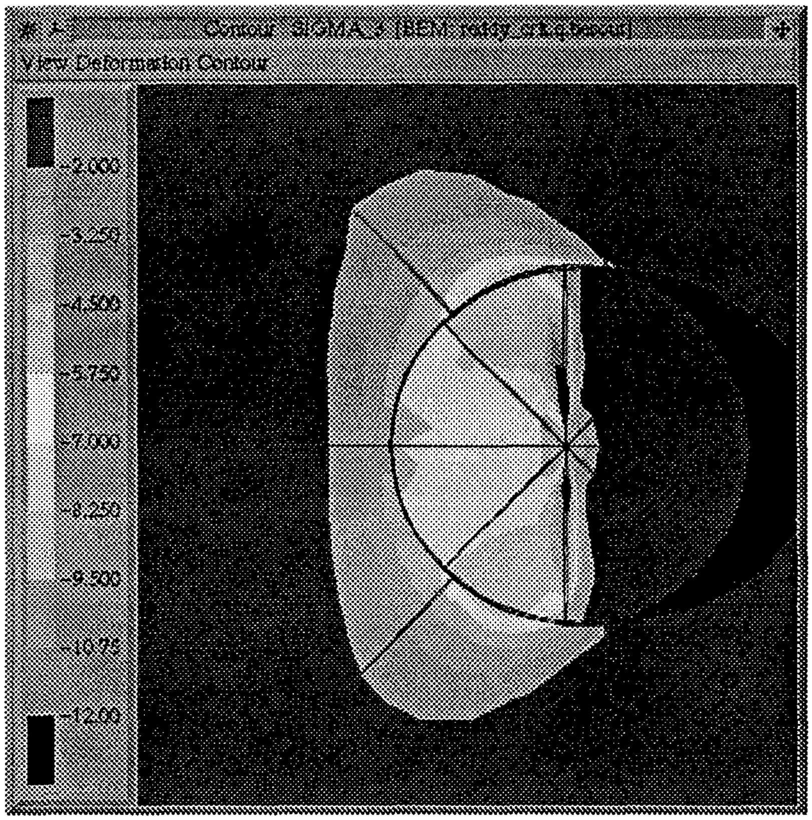


Figure 2-8. Color contours of the minimum principal stress also shows that the maximum compressive contact stresses are at about 65 degrees. Magnification factor is 100.

The minimum value of normal traction are substanitally less than the experimental experimental values reported Joh (about -35 ksi). It is unclear from the Yogeswaren and Reddy paper whether the reported load value of 1210 lbs. was the total load applied to the experiments, or the total load applied to their finite element mesh, which models one half of the experiment. In retrospect, it appears that the later is the case, while we assumed that the reported load was the total experimental load, and only applied 605 lbs. to our half symmetry analysis. Furthermore, Rao [1978] has shown that the amount of clearance and friction can significantly effect the contact stress distribution and location of separation. Therefore, direct comparisons between our analysis and the experiments by Joh cannot be made.

A second model was analyzed with a stiffer "steel" pin (E=30,000 ksi). The normal traction on the contact surface for this model is shown in Figure 2-9. The trend of the curve is similar to the aluminum pin, although the minimum is shifted to about 60 degrees.

In both cases, the minimum normal traction on the contact surface is not at position A (0 degrees), because the pin is close to the fixed end of the plate. A third model was created by extending the plate 3" to the left, leaving the pin much further from the fixed end. The steel pin was used in this model. The normal traction on the contact surface for this model (Figure 2-9) shows that the normal traction is almost constant from 0 to about 60 degrees after which it decays to zero.

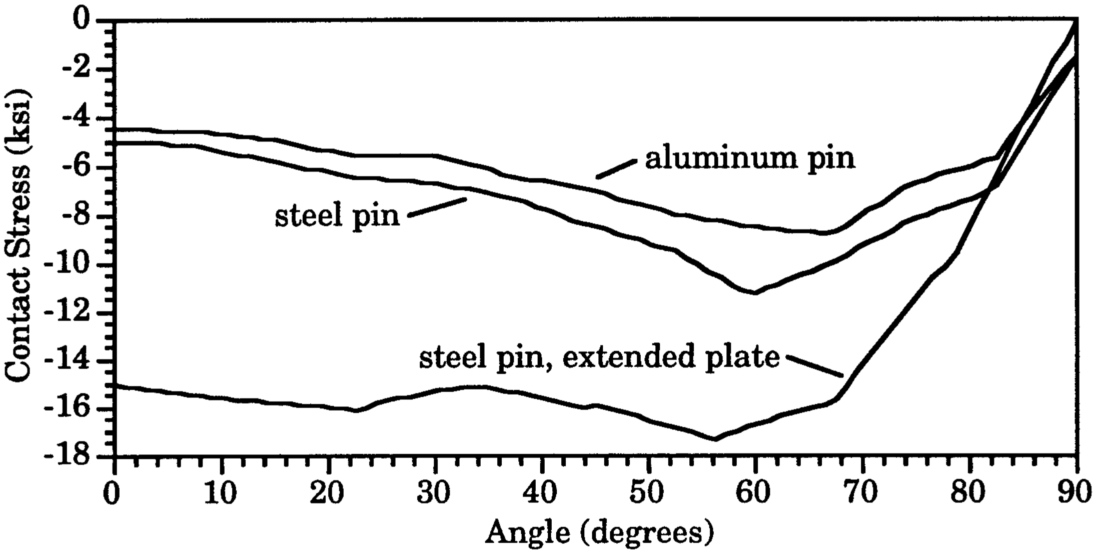


Figure 2-9. Contact normal stress for an aluminum pin, steel pin, and a steel pin with an extended plate.

3. Additional Tasks not in the Workstatement

There were three tasks not in the workstatement of our Phase I proposal for which we spent a considerable amount of effort during the course of this project. In fact, these efforts represent the majority of our effort.

First, we extended our two-dimension symmetric-Galerkin BEM code to include single domain fracture analysis, for orthotropic materials. Second, in anticipation of three-dimensional developments, we formulated an efficient technique for evaluating 3D anisotropic Green's functions. Finally, we have formulated and implemented a three-dimensional anisotropic Galerkin (but not yet symmetric) BEM code.

3.1 2D Symmetric Galerkin Fracture Mechanics

A symmetric Galerkin algorithm for fracture, in the context of two-dimensional potential theory, was recently presented by Gray, Balakrishna, and Kane [1995]. This method, effectively a combination of displacement discontinuity [Crouch and Starfield 1983] and the hypersingular BEM [e.g., Gray 1990] ideas, employs the hypersingular equation on the crack surface as described above in Section 1.3. For the present project, we have explored techniques to improve the computational efficiency, and have extended the approach to plane orthotropic elastostatics. Some of this work has been presented in Gray and Paulino [1996].

3.1.1 Formulation

Consider a 2D, linear-elastic, orthotropic body which contains a crack, as illustrated in Figure 3-1. The body has a boundary $\Gamma = \Gamma_u + \Gamma_\tau + \Gamma_c^+ + \Gamma_c^-$, which represents, respectively, the portion of the boundary over which displacements are specified, tractions are specified, and the two sides of the crack (over which the tractions are assumed to be known).

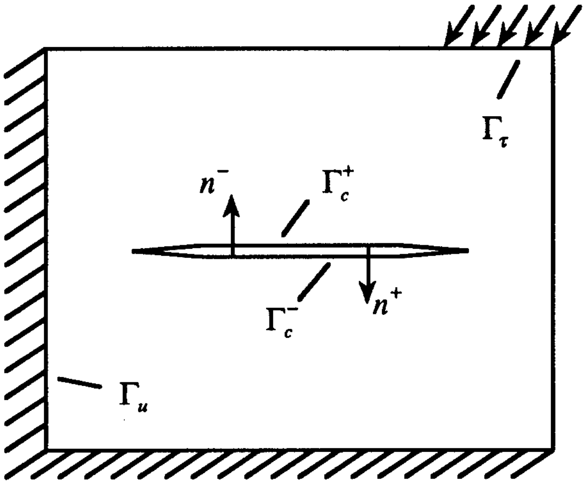


Figure 3-1 A generic body containing a crack.

In the hypersingular approach to solving fracture problems described in Section 1.3, independent equations for solving for the displacements on both sides of the crack are obtained by writing both standard and hypersingular equations. This violates the symmetric-Galerkin procedure, as the boundary condition on the crack surfaces are assumed to be specified tractions and, consequently, only the traction equation should be used. Symmetry of the coefficient matrix is therefore not possible with the hypersingular approach. Nevertheless, taking this as a starting point, the resulting system of equations can be written in block-matrix form,

$$\begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{Bmatrix} \Omega_1 \\ u_2 \\ u_3 \end{Bmatrix} = \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \end{bmatrix} \begin{Bmatrix} \Psi_1 \\ \tau_2 \\ \tau_3 \end{Bmatrix}. \tag{3-1}$$

The blocking strategy, according to Eq. (3-1), is as follows: the first row and column is associated with the outer, non-crack boundary $\Gamma_u + \Gamma_\tau$, while subscripts 2 and 3 refer to the two sides of the crack Γ_c^+ and Γ_c^-, respectively. The vector of unknowns on the outer boundary is, in general, a mixture of displacements and tractions, and is denoted by Ω_1. The corresponding vector of prescribed boundary values is indicated by Ψ_1. The second and third rows correspond to the equations for traction and displacement, respectively, written on the crack surface (again, it assumed that the traction is specified on the crack surface).

Note that, although the coefficient matrix in Eq. (3-1) is not symmetric, the upper 2 x 2 principal submatrix is, *i.e.* $H_{11} = H_{11}^T$, $H_{22} = H_{22}^T$, and $H_{21} = H_{12}^T$. This is a consequence of the basic symmetric-Galerkin procedure, the traction being the appropriate equation choice on the crack. In addition to the symmetry, it is also important to observe that $H_{31} = -H_{12}$ and $H_{23} = -H_{22}$, a consequence of the change in surface orientation in the integration over the two sides of the crack. It can be shown that a symmetric coefficient matrix results from changing variables on the crack from displacement to the jump in displacement (*i.e.,* displacement discontinuity),

$$\begin{aligned} \Delta u &= u_2 - u_3 \\ \Sigma u &= u_2 + u_3 \end{aligned}. \tag{3-2}$$

With this transformation, the left hand side of Eq. (3-1) takes the form

$$\begin{bmatrix} H_{11} & H_{12} & 0 \\ H_{12}^T & H_{22} & 0 \\ H_{31} & H_{32} & I/2 \end{bmatrix} \begin{Bmatrix} \Omega_1 \\ \Delta u \\ \Sigma u \end{Bmatrix}, \tag{3-3}$$

where I is the identity matrix. It therefore suffices to solve the smaller symmetric 2 x 2 block system for the unknowns $\{\Omega_1, \Delta u\}$. If necessary, Σu, and hence the actual crack face displacements can be calculated in a post-processing step.

3.1.2 Example Problems

Two examples are presented to validate the 2D, orthotropic, symmetric-Galerkin fracture capability. Each of these examples involves a single interior crack, one aligned with the material axes (Figure 3-2a) and one inclined at a 45° to the axes (Figure 3-2b). For both examples, plane stress conditions are assumed. The crack has length 2a = 0.4, the plate has height 2H = 2.0 and width 2W = 1.0. A remote traction of 10.0 MPa was applied. The elastic constants used, which correspond to average homogenized properties for fiberglass, are [Ghandi 1972]:

$$\begin{aligned} E_1 &= 48.26\ \text{GPa} \\ E_2 &= 17.24\ \text{GPa} \\ \mu_{12} &= 6.89\ \text{GPa} \\ \nu_{12} &= 0.291 \end{aligned}$$

The boundary element results for these examples are compared to finite element solutions from the program FRANC2D [Wawrzynek 1991, Boone, Wawrzynek, and Ingraffea1987]. The parameter chosen for comparison purposes is the displacement discontinuity Δu along the crack.

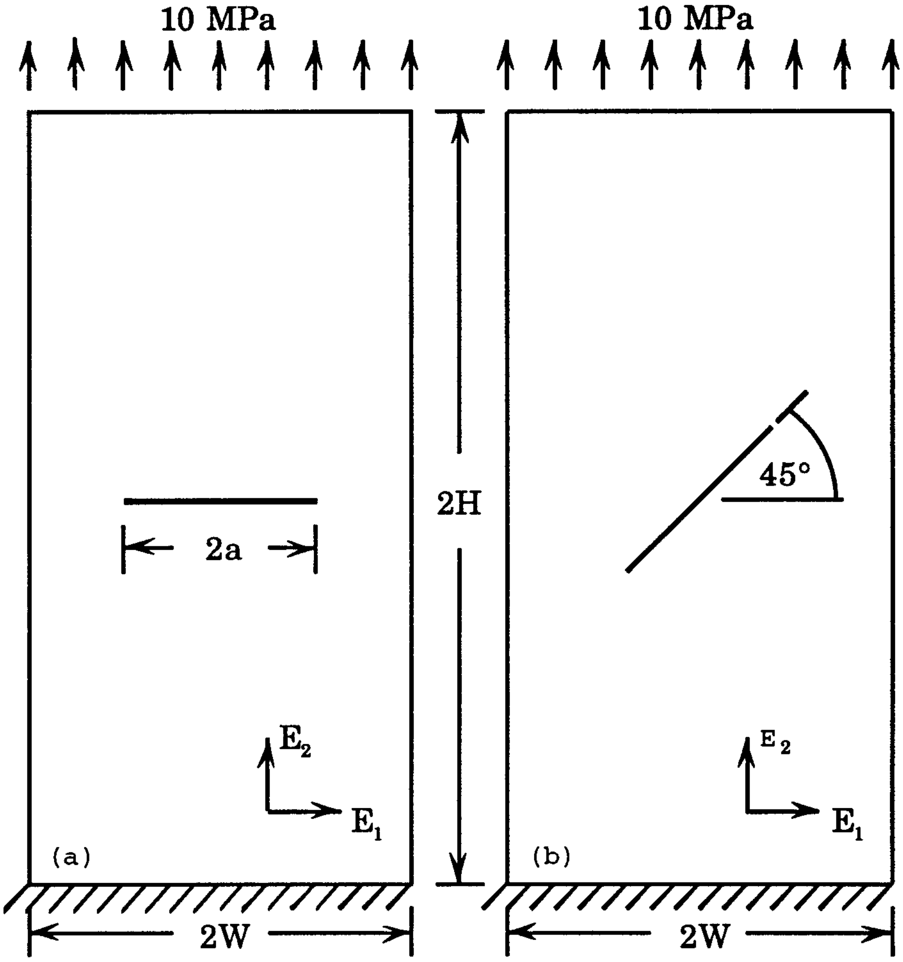


Figure 3-2. The two example problems used to validate the 2D, symmetric-Galerkin boundary element capability.

The finite element solutions used approximately 270 quadratic order elements with about 1560 degrees-of-freedom. Quarter-point elements are used to model the stress singularity at the crack tips.

For the boundary element solutions, the outer boundary of the plate has been uniformly discretized with 60 standard *linear* elements. Each face of the crack has been discretized with 8 linear boundary elements, with some grading towards finer elements towards the crack tip. Such discretization, especially of the crack faces, is much too coarse to expect highly accurate results, especially in the crack tip region where displacement gradients are high. However, it was felt that if meaningful comparisons to the FEM results could be made, this would be sufficient to validate the approach.

Table 3-1 shows the results for the crack opening displacement (COD) for the crack of example 1. Because of symmetry, only the results for the left portion of the crack are computed. The COD's obtained by both the BEM and FEM analyses are in reasonably good agreement, the largest discrepancy being less than 5%. For this problem, the crack sliding displacement (CSD) should be zero. The numerical values for both BEM and FEM are of the order 10^{-8} or less, indicating that consistent solutions have been obtained.

Table 3-1. Crack opening displacement for Example 1. The solution is for one half of the crack starting at the left crack tip.

x	y	SG-BEM Δu_y ($\times 10^{-3}$)	FEM Δu_y ($\times 10^{-3}$)	difference %
0.300	1.000	0.000000	0.000000	0.00
0.325	1.000	0.227131	0.230606	1.51
0.500	1.000	0.391940	0.411062	4.65
0.450	1.000	0.439092	0.460448	4.64
0.500	1.000	0.453433	0.476836	4.91

Table 3-2 shows the results for the displacement discontinuity for the entire lenght of the crack of example 2. As before, the results for Δu_y obtained by both the BEM and FEM are in reasonable agreement, the largest discrepancy being about 5%. However, the results for Δu_x show much larger relative discrepancies, especially for the nodes near the crack tips. It is quite likely that for this case neither the BEM or FEM solutions are converged. However, for the present purposes, these analyses demonstrate that our extensions of the symmetric Galerkin boundary element formulation for fracture problems is valid.

Table 3-2. The displacement discontinuities for Example 2.

x	y	BEM		FEM		Δu_x (%)	Δu_y (%)
		Δu_x ($\times 10^{-3}$)	Δu_y ($\times 10^{-3}$)	Δu_x ($\times 10^{-3}$)	Δu_y ($\times 10^{-3}$)		
0.359	0.859	0.000000	0.000000	0.000000	0.000000	0.00	0.00
0.376	0.876	-.049903	0.163476	-.065051	0.164245	23.29	0.47
0.429	0.929	-.105063	0.283705	-.113276	0.298861	7.25	5.07
0.465	0.965	-.125796	0.318479	-.139346	0.334257	9.72	4.72
0.500	1.000	-.132743	0.329145	-.145958	0.345614	9.05	4.77
0.535	1.035	-.125730	0.318613	-.139309	0.334217	9.75	4.67
0.571	1.071	-.104960	0.283939	-.113400	0.298794	7.44	4.97

0.624	1.124	-.049838	0.163708	-.064930	0.164171	23.24	0.28
0.641	1.141	0.000000	0.000000	0.000000	0.000000	0.00	0.00

3.2 Effective Evaluation of the 3D Anisotropic Green's Function

3.2.1 Formulation

A necessary component of a practical program for performing 3D BEM stress analysis for orthotropic materials is a routine for efficiently evaluating the anisotropic Green's function. This function, which can be stated as:

$$U(x,y) = \frac{1}{8\pi^2 r} \oint_{S^1} K^{-1}(\xi)\, ds(\xi) \equiv \frac{1}{8\pi^2 r} \tilde{U}, \tag{3-4}$$

where the Christoffel matrix, K, is defined in terms of the elastic constants, C, of the material,

$$K_{ij} = \sum_{l,m} C_{iljm} \xi_l \xi_m, \quad \text{and,} \quad r = \|x - y\|. \tag{3-5}$$

The integration path is the unit circle in the plane that is perpendicular to the vector $x - y$,

$$S^1 = S^1(x,y) = \left\{ \xi \in \Re^3 \mid \|\xi\| = 1,\, \xi \cdot (x-y) = 0 \right\}. \tag{3-6}$$

Formulation of BEM equations requires numerous Green's function evaluations, and thus direct numerical quadrature is simply too computationally expensive.

Previously, the best available algorithm has been the method proposed by Wilson and Cruse, [1978]. In their approach, reasonable computation time was obtained by employing cubic interpolation with tabulated precalculated values. This method has three potential shortcomings. First, storage requirements can become significant, especially if first and second derivatives of the Green's function are required (such as for evaluation of hypersingular equations used for fracture mechanics). Second, as noted in Schclar [1994], accuracy may be a concern, as the variation in the function values for a highly anisotropic material is likely to be significantly larger than for a mildly anisotropic solid, and thus the number of precalulated points needed to maintain accuracy is a function of the anisotropy of the material. Third, the precalculated points are material specific. The precalculation process must be performed, and the resulting tables must be maintained, for each material of interest.

For this project we decided to implement the residue calculus technique for evaluating the function described by Sales and Gray [1996]. This approach gives us the obvious benefit of the direct participation of one of the authors, access to the software developed for the paper, and capability to develop a subroutine that would work for all orthotropic materials with no precalculations or storage of tables required. The residue approach has also been exploited by Wang [1995]; however, this paper is primarily theoretical, and does not focus on computations.

Following Sales and Gray, the contour integral above is written as

$$\tilde{U}(\theta,\psi) = \int_0^{2\pi} K^{-1}(\xi(t))\, dt, \tag{3-7}$$

where ξ is written parametrically as

$$\xi(t) = \begin{bmatrix} \xi_1(t) \\ \xi_2(t) \\ \xi_3(t) \end{bmatrix} = \begin{bmatrix} \sin(\theta)\cos(t) + \cos(\theta)\cos(\psi)\sin(t) \\ -\cos(\theta)\cos(t) + \sin(\theta)\cos(\psi)\sin(t) \\ -\sin(\psi)\sin(t) \end{bmatrix} \tag{3-8}$$

The integration can be performed using residue calculus by transforming the integrand into a rational function and expanding the range of the integral to include all real numbers. As shown by Dederichs and Leibfried [1969], this can be done with the substitution

$$Z = \tan(t). \tag{3-9}$$

With this substitution, ξ can be written as

$$\xi = \cos(t)s(Z), \tag{3-10}$$

where

$$s(Z) = \begin{bmatrix} \sin(\theta) + Z\cos(\theta)\cos(\psi) \\ -\cos(\theta) + Z\sin(\theta)\cos(\psi) \\ -Z\sin(\psi) \end{bmatrix}. \tag{3-11}$$

Since the elements of ξ are linear functions of Z, and K is a quadratic function of ξ, the elements of K are second order polynomials in Z. Thus, by Cramer's rule, the elements of K^{-1} are rational functions of Z,

$$K_{jk}^{-1} = \frac{1}{\cos^2(t)} \frac{P_{jk}(Z)}{Q(Z)}, \tag{3-12}$$

where P and Q are polynomials of Z,

$$P_{jk}(Z) = \sum_{l=0}^{4} a_{jkl} Z^l, \tag{3-13}$$

$$Q(Z) = \sum_{l=0}^{6} b_l Z^l. \tag{3-14}$$

With the observation that $1/\cos^2(t) = dZ/dt$, the Green's function can be written in the form

$$\tilde{U}_{jk} = 2\int_{-\infty}^{\infty} \frac{P_{jk}(Z)}{Q(Z)} dZ \tag{3-15}$$

$$= 4\pi i \sum_{n=1}^{3} Residue_{Z=\lambda_n} \frac{P_{jk}(Z)}{Q(Z)} \tag{3-16}$$

$$= 4\pi i \sum_{n=1}^{3} \frac{P_{jk}(Z)(Z-\lambda_n)}{Q(Z)} |_{z=\lambda_n} \tag{3-17}$$

$$= 4\pi i \sum_{n=1}^{3} \frac{P_{jk}(\lambda_n)}{Q(\lambda_n)} \tag{3-18}$$

where λ_1, λ_2, and λ_3 represent the three roots of Q(Z) in the upper half of the complex plane.

To evaluate the Green's function numerically using this technique is a two step process. First, the three roots, λ_n, are determined using Newton's method. Second, Eq. 3-18 is evaluated.

The coefficients of *P* and *Q* (a_{jkl} and b_l in 3-13 and 3-14) are functions of the components of K^{-1}, and thus depend on the material constants. For orthotropic materials there are nine independent material constants (c_{11},c_{12}, ... c_{66}). For the current project, the Maple V symbolic algebra program was used to construct the *K* matrix symbolically in terms of the nine material constants. The matrix was then symbolically inverted, and expressions for the coefficients of *P* and *Q* were developed. While these expressions are rather tedious, the Maple program is capable of automatically generating FORTRAN code for their evaluation.

A similar technique can be used to determine first and second derivatives of the Green's function. This process is detailed in the paper by Sales and Gray, and will not be repeated here.

3.2.2 Testing

We have performed a series of tests to assess the correctness and accuracy of the new software. For example, consider the evaluation of the normalized Green's function along vectors lying in the a plane passing through the y axis and making a 45° angle with the x and z axis, as shown in Figure 3-3.

We have evaluated the Green's function using both our software, and a direct numerical integration using Simpson's rule (for 60 segments). A plot of the relative percentage error for two of the components is shown in Figure 3-4.

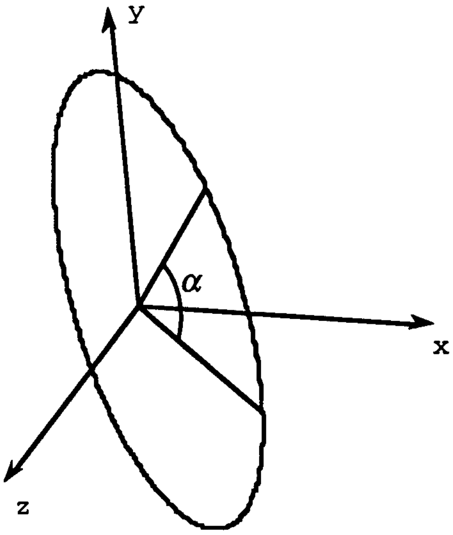


Figure 3-3. Path for the evaluation of Green's function

As can be seen by this graph, the accuracy of this approach is very good, with maximum errors less than 1/100th of a percent.

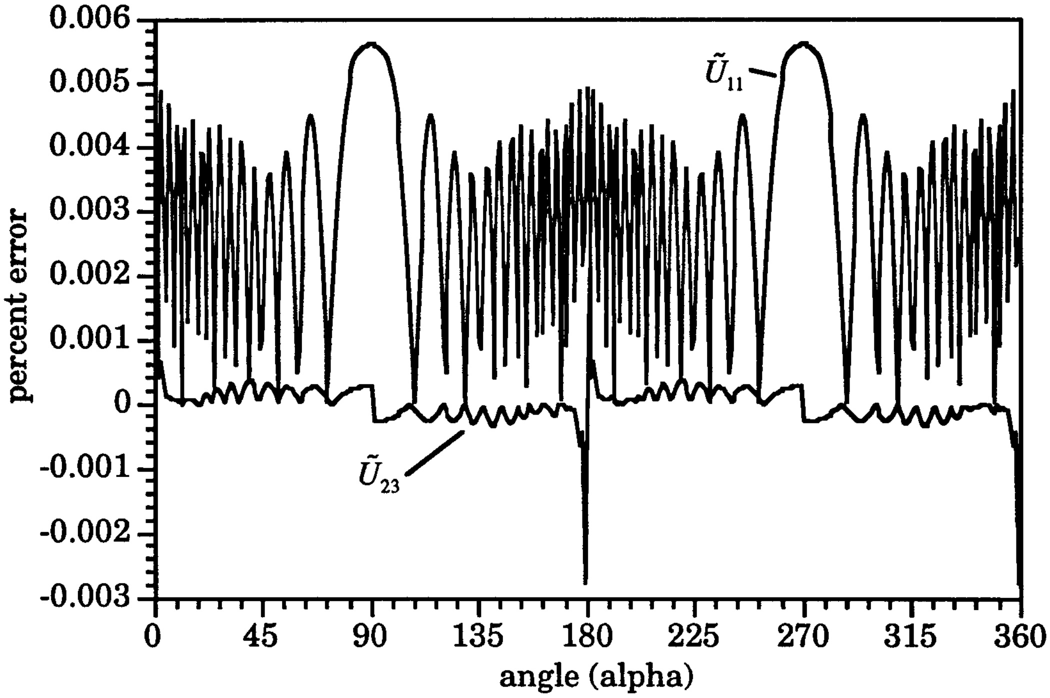


Figure 3-4. Relative percent error for two components of the Green's function

3.3 Development of a 3D Symmetric Galerkin BEM Code

A substantial portion of the resources for this project were spent developing a new, 3D, symmetric-Galerkin code and interfacing it with our FRANC3D modeling/meshing/postprocessing system. This work is forward-looking towards Phase II, where it is anticipated that the symmetric-Galerkin formulation will provide an efficient boundary element technique that can accurately determine stresses and displacements near cracks and other boundaries.

3.3.1 Code Organization and Development

There are three major tasks that a BEM program is required to perform: read input data and precompute heavily reused data, perform all element integrations, and solve the resulting system of simultaneous equations. The software in our new program is organized along these lines.

Read Input and Precompute Element Data

The subroutines that read the input file and precompute element information are written in the C programming language. We feel that C language's capacity for handling dynamic memory allocation and for manipulating character based information makes it an ideal choice for these routines. In particular, because of the dynamic memory management the language provides, there is no need to "preset" limits on the number of nodes, elements, and materials that the program can handle. Such limits will be determined at runtime based on the available resources.

With a view towards flexibility and computational efficiency, there are two concepts that we exploit quite heavily in the code: generalized element data, and precomputation of this data. By generalized element data we mean that much of the information that one needs on an element basis has been formulated in a manner that will work for all linear and quadratic order elements. For example, one often needs to determine the Cartesian coordinates for a point in an element given in parametric coordinates. This is easily and conventionally done by using the element shape functions to interpolate the element's nodal coordinates. That is, symbolically,

$$x_\alpha = \sum_{i=1}^{nnodes} N(u,v) x_{i\alpha}, \quad \alpha = 1..3. \tag{3-19}$$

Computationally, this procedure becomes more involved. One must first compute the values of the shape functions at the specified parametric coordinates. These will be different and of different number for each type of element. This leads to case analyses illustrated in the following FORTRAN code:

```
      if (nnodes .eq. 3) then
          call T3_shape(u,v,shape)
      else if (nnodes .eq. 4) then
          call Q4_shape(u,v,shape)
      else if
          ....
      end if

      do j = 1,3
          coord(j) = 0.0
          do i = 1,nnodes
              this_node =
     1          node_coords(j,connectivity(i,this_elem))
              coord(j) = coord(j) + shape(i)*this_node
          end do
      end do
```

Such case analyses are undesirable for two reasons. First, if a new element is added to the program (*e.g.*, a new crack tip element), all such case analyses must be found and modified. Second, such case analyses can seriously hinder or defeat a compiler's effort to optimize, vectorize, or parallelize the code.

In our code, this evaluation is generalized for all element types using the expression,

$$x_\alpha = a_{\alpha 1} + a_{\alpha 2}u + a_{\alpha 3}v + a_{\alpha 4}u^2 + a_{\alpha 5}uv + a_{\alpha 6}v^2 + a_{\alpha 7}u^2v + a_{\alpha 8}uv^2, \quad \alpha = 1..3. \quad (3\text{-}20)$$

The corresponding FORTRAN code is thus,

```
      u2 = u * u
      v2 = v * v
      uv = u * v
      u2v = u2 * v
      uv2 = u * v2

      do j = 1,3
          coord(j) = a(j,1) + a(j,2)*u + a(j,3)*v +
     1               a(j,4)*u2 + a(j,5)*uv + a(j,6)*v2 +
     1               a(j,7)*u2v + a(j,8)*uv2
      end do
```

It may appear that this computation would be expensive for the simpler elements, where many of the *a* coefficients would be zero. Actually, if one accounts for the computation of the shape function, and the loop and conditional overhead, the latter is only slightly more expensive in terms of operations for these elements, and significantly cheaper for the higher order elements.

This example is a simplification of the actual situation in the code. There we need to be able to evaluate the coordinates for point in terms of a polar expansion (ρ,θ) about any location in the element. The resulting expression is,

$$
\begin{aligned}
x_\alpha = {} & \left(a_{\alpha1} + a_{\alpha2}u + a_{\alpha3}v + a_{\alpha4}u^2 + a_{\alpha5}uv + a_{\alpha6}v^2 + a_{\alpha7}u^2v + a_{\alpha8}uv^2\right) + \\
& \left[\begin{array}{l} \left(a_{\alpha9} + a_{\alpha10}u + a_{\alpha11}v + a_{\alpha12}u^2 + a_{\alpha13}uv + a_{\alpha14}v^2\right)\cos(\theta) + \\ \left(a_{\alpha15} + a_{\alpha16}u + a_{\alpha17}v + a_{\alpha18}u^2 + a_{\alpha19}uv + a_{\alpha20}v^2\right)\sin(\theta) \end{array}\right]\rho + \\
& \left[\begin{array}{l} \left(a_{\alpha21} + a_{\alpha22}u + a_{\alpha23}v\right)\cos^2(\theta) + \\ \left(a_{\alpha24} + a_{\alpha25}u + a_{\alpha26}v\right)\cos(\theta)\sin(\theta) + \\ \left(a_{\alpha27} + a_{\alpha28}u + a_{\alpha29}v\right)\sin^2(\theta) \end{array}\right]\rho^2 + \\
& \left[\left(a_{\alpha30}\right)\cos^2(\theta)\sin(\theta) + \left(a_{\alpha31}\right)\cos(\theta)\sin^2(\theta)\right]\rho^3
\end{aligned}
\tag{3-21}
$$

There are two classes of precomputed data, those which are done for each individual element, and those which are done for each element *type*.

The following data is precomputed in the program:

for each element

1. The Cartesian coordinates of all Gauss integration points,
2. The integration weights for all Gauss points,
3. The element normal for all Gauss points,
4. The coefficients needed to find the Cartesian coordinate for any location in an element given the parametric coordinates of an origin point and polar offsets from that origin (Eq. 3-21), and
5. The coefficients needed to evaluate the Jacobian at any location in an element given the parametric coordinates of the point.

for each type of element

1. The parametric coordinates of the Gauss integration points,
2. The integration weights for all Gauss points,
3. The shape function values at all Gauss points,
4. The beginning and ending angle and the coefficients needed to find the distance to the element edge, for a polar integration over the element area, centered at each corner node,
5. The coefficients needed to find the values of the shape functions at any location in the element given the parametric coordinates of an origin point and polar offsets from that origin, and
6. The parametric coordinates of the corner nodes of the element.

By precomputing all this information, we have completely avoided any case analysis inside any of the element integration routines. Furthermore, since much of this information is heavily reused during the integration process, we save the cost of recomputing the information every time it is needed.

Element Integrations

Much of the program is devoted to computing element integrations. This portion of the program is written in FORTRAN. FORTRAN is the conventional programming language for such computationally intense algorithms, and, in general, the best optimizing and parallelizing compilers available are for FORTRAN code.

There are four different cases of element integrations depending on the physical locations of the source and destination elements. These are illustrated in Figure 3-5. In all cases the integrands involve $1/r$ or $1/r^3$ terms, leading to singular integrals in three of the four cases.

Most integrations are of the *Disjoint* type (Figure 3-5a). This is a nonsingular integration. *Coincident* integrations (Figure 3-5b) occur when the source and destination element are the same. This integral becomes singular as the Q point approaches the P point. There are two possible types of *adjacent* integrations. When the two elements share a common edge (Figure 3-5c) the integrand is singular along the entire common edge. When the two elements share a common vertex (Figure 3-5d) the integrand is singular only at this vertex,

Within the code, different computational strategies are employed for each of the four cases. The approaches are summarized here and described in detail in Appendix A.

Disjoint

This case is relatively straightforward and is handled with conventional Gauss integration. The integrand contains products of the element shape function values, Jacobean determinates, Gauss integration weights, and appropriate kernel functions.

Coincident

The integrand in this case becomes singular as the source point approaches the destination point. To handle this situation, the integrand is decomposed into singular and nonsingular parts. If for each gauss point a transformation is made to a polar coordinate system centered at the

Gauss point, an analytical expression for the singular part of the integrand can be found.

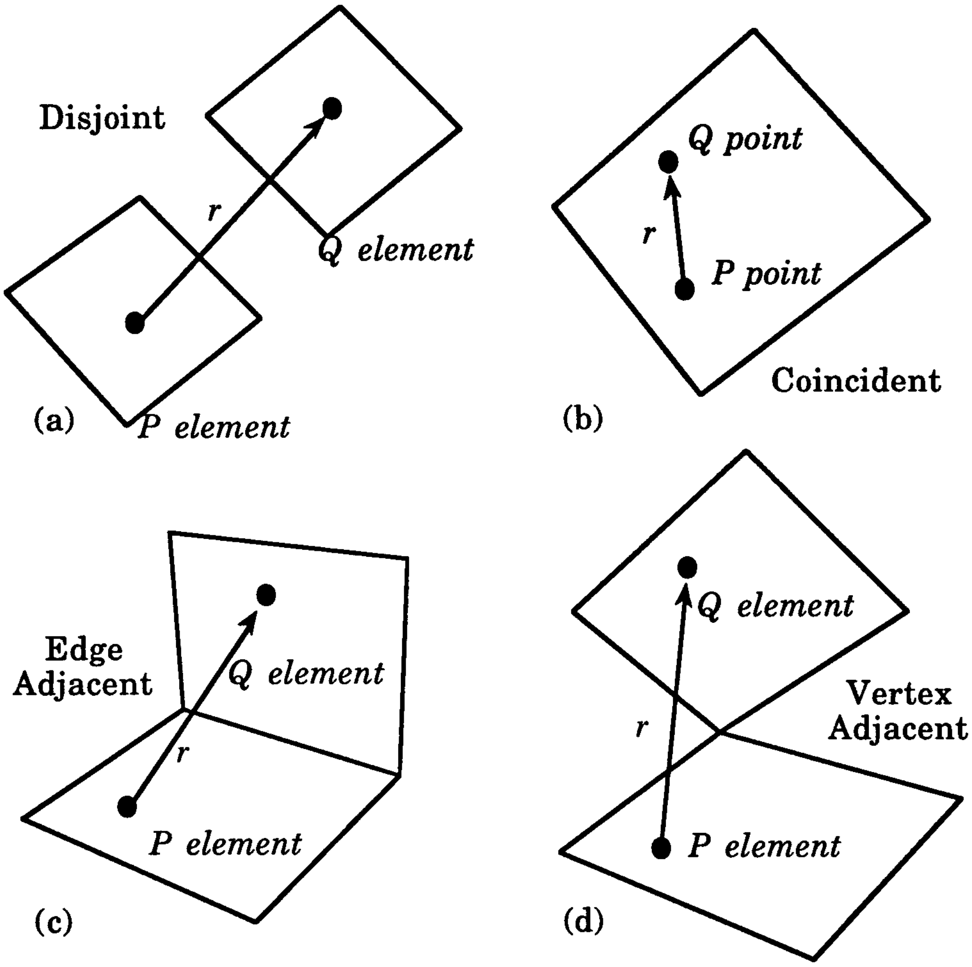


Figure 3-5 The four possible configurations for element integrations.

The first step is to break the element into a number of triangles, $T_{1..4}$ as shown in Figure 3-6 (the description given here is for quadrilateral elements, but works similarly for triangular elements). Within each triangle, the integration over θ is performed numerically (that is, discrete angles and weights are chosen based on standard quadrature rules). The integration over ρ, however is performed analytically. This process is repeated for each of the Gauss integration points in the element.

Vertex Adjacent

The integrand in this case becomes singular at the common vertex. To handle this situation, the integrand is again decomposed into singular and nonsingular parts. Again, an analytical expression for the singular integral can be found for a polar coordinate system, in this case, centered at the common vertex.

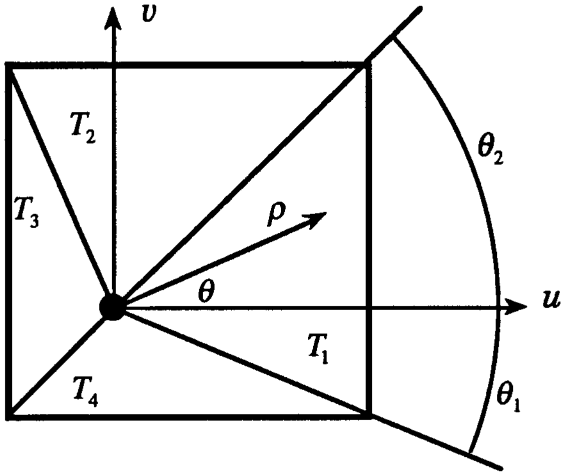


Figure 3-6 Coincident polar element integration.

This situation is illustrated in Figure 3-7. The Q element is decomposed into two triangles, T_1 and T_2 (only one triangle is necessary for triangular elements). Within each triangle, the integration over θ_q is performed numerically. The integration over ρ_q, is partly analytic and partly numerical.

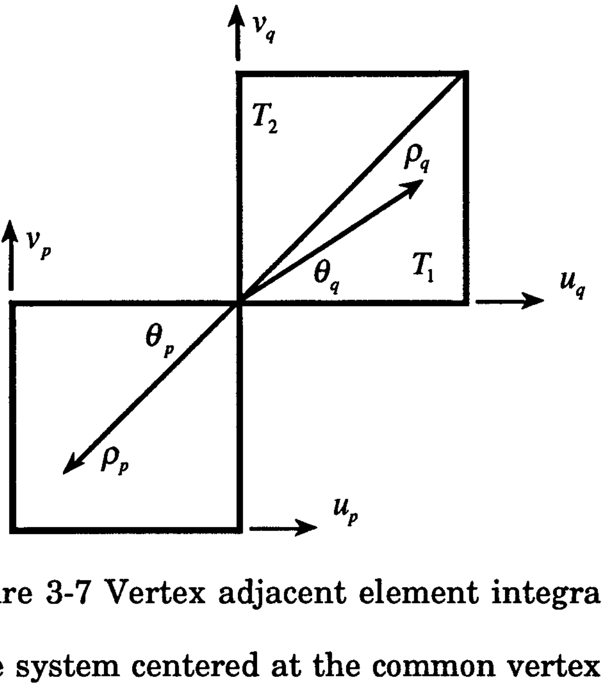


Figure 3-7 Vertex adjacent element integration.

A polar coordinate system centered at the common vertex is used in the P element also. However, in this case, both the θ_p and ρ_p integrations are done numerically by using the standard Gauss integration points and weights (and finding the appropriate θ_p and ρ_p that correspond to these points).

Edge Adjacent

The edge adjacent case is similar to vertex adjacent, but now the singularity is along the entire common edge. The integrand is again decomposed into singular and nonsingular parts. In this case, however, we use mixed coordinate systems.

As illustrated in Figure 3-8, a polar coordinate system is used for the *Q* element. The origin of the coordinate system is a point on the common edge. This point is the projection of the *P* integration point onto this edge, and thus moves for the different Gauss points in the *P* integration.

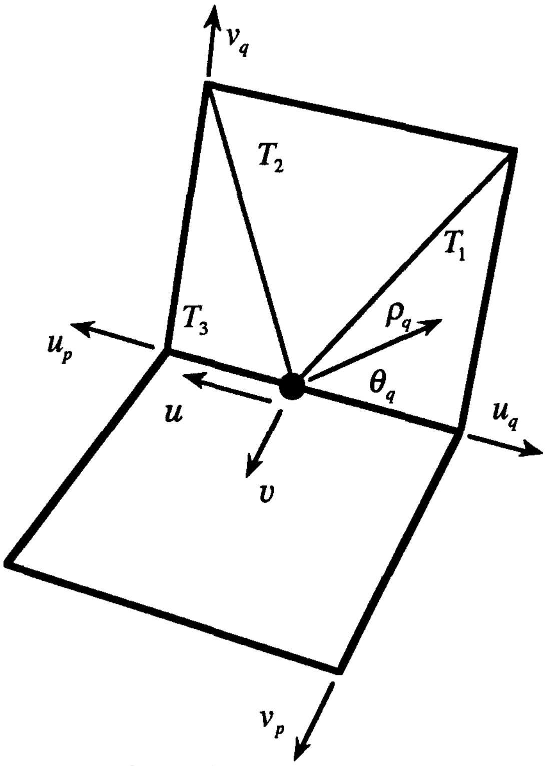


Figure 3-8. Edge adjacent element integration.

The *Q* element is decomposed into triangles, two for triangular element and three for quadrilaterals. Within each triangle, the integration over θ_q is performed numerically. The integration over ρ_q, is performed analytically.

The *P* element integration is performed in a rectilinear coordinate system. The integration for the *u* coordinate is performed numerically; however, an analytical expression has been derived for the integration along the *v* coordinate.

Our algorithm for treating the edge adjacent case is a new development. The resulting analytical expressions are complicated, but are easily derived using symbolic computations.

Solving Systems of Equations

Once element integrations have been performed, the resulting system of simultaneous linear equations must be solved. While this procedure is computationally expensive, it is also well understood, and many linear equation solves are available. Our intention is to not spend any more effort on this part of the program than necessary.

For testing, where the coefficient matrix is small enough to be contained in virtual memory without excessive thrashing, we are using the widely available LINPAC solver.

When we start working with realistic problems, with correspondingly larger coefficient matrices, we anticipate using the out-of-core BEM optimized solver that we developed for our existing BEM program.

3.3.2 Integration with FRANC3D

We have modified the FRANC3D program to generate input files for the new BEM code. This effort was more involved than originally anticipated because the new code treats "corner" boundary conditions differently than our existing BEM code.

In the new program, "ambiguous" corner boundary conditions are treated by duplicating the appropriate nodes. For each degree-of-freedom at each node, either a traction or displacement can be specified, with the other being an unknown. At corners, however, traction values are not necessarily unique, leading to a possibility of more than one unknown at the corners. If this is the case, we need to "duplicate" the node so that the number of knowns matches the number of unknowns. Duplicating the node affects the node list and the nodal connectivity table, thus complicating input file generation.

This situation can be illustrated in 2D. Figure 3-9 shows (a) a corner and (b) a non-corner configuration. First assume that the displacement at the node is known. In case (b), where n_1 equals n_2, t_1 equals t_2, we have one unknown (traction) for our known (displacement). In case (a), however, the normals and corresponding tractions will be different. Therefore, we have two unknown (tractions) for one known (displacement). In this case we must duplicate the node so that both tractions appear as unknowns in the system.

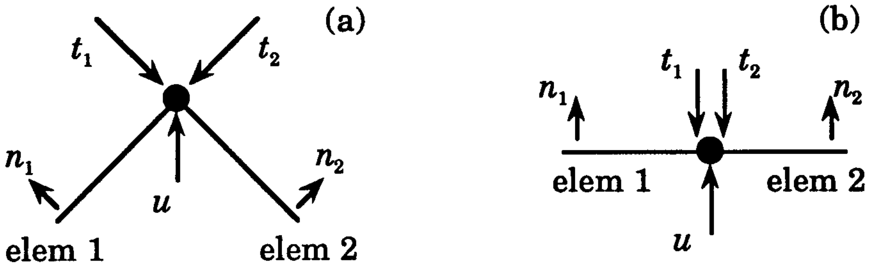


Figure 3-9. 2D corner and non-corner boundary condition configurations.

If, on the other hand, tractions are known at the corner, there are again two possible cases. If the tractions are equal then we have one known and one unknown. If, however, the tractions are not equal, there are two knowns for one unknown, and we must duplicate the node.

The 3D case is a generalization of this 2D analysis, but is significantly complicated by the fact that there may be any number of elements incident at a node, each with their own known and unknown boundary condition type and value. In the 3D case, the actual number of nodes created will be between one and the number of incident elements.

To determine the number of actual nodes created in the input file for any node location, the following general algorithm was developed:

1. Assume initially the worst case that one node will be generated for each incident element. We generate a data structure for each such "use" of a node, which contains the types of known and unknown boundary conditions, the known boundary condition values, and the element's normal.

2. Compare all possible combinations of node uses and determine which node uses have equivalent knowns and unknowns, and can be grouped together.

3. Two node uses are determined to be equivalent if they meet the following criteria:

   a. They both have the same types of known boundary conditions for all three coordinate degrees of freedom.
   b. If the tractions are known, then all three traction values must be equal.
   c. If the displacement is known, then the element normals must be equal.

4. One node is generated in the input file for each group of element uses, and the appropriate element nodal connectivity is generated.

The above algorithm has been implemented and tested within FRANC3D, and appears to be working well.

The remainder of the BEM input file creation is a relatively straight forward specification of nodal coordinates, element connectivity, material properties, and boundary conditions. The only significant additional consideration is the case of bi-material interfaces. For the elements on the interface, the BEM code expects to see two elements in the input file, with reversed nodal ordering (each ordering corresponds to an outward pointing normal for the two materials) and differing material numbers. This is easily handled by way of the FRANC3D internal data structures.

4. Summary and Conclusions

We have successfully completed a six month Phase I project to show the feasibility of developing a 3D boundary element analysis system for composite joints with discrete damage. Our overall goal for Phase I was to demonstrate expertise in BEM methods appropriate to the detailed stress analysis of arbitrary bolted or bonded composite joints containing cracks and contact surfaces.

We have completed all three tasks in the Phase I workstatement:

- We augmented the Galerkin-based, 2D, orthotropic BEM program developed for the original program solicitation to include *hypersingular* BEM traction equations so that a symmetric coefficient matrix could be formulated. We believe that this orthotropic symmetric-Galerkin code is unique.
- Using the code developed in Task 1, we solved an example laminate problem [Pagano 1978], to show that the symmetric-Galerkin (SG) approach can yield accurate interlaminar stress predictions near free surfaces.
- We demonstrated a capability to perform 3D contact analysis using our existing 3D BEM program. The approach is a penalty function technique derived from a current capabilities for fluid driven fracture, and can also be used for the proposed Phase II 3D simulation system.

In addition to these tasks, considerable resources were spent on three additional tasks that further our goal of demonstrating expertise, and are preparatory for a potential Phase II effort.

- We have added fracture analysis capabilities to the 2D SG code of Task 1. We believe that this is the first SG fracture for elasto-statics, and certainly the first for orthotropic materials.

- We have developed software to evaluate the 3D anisotropic Green's function. This is a generalization of the work of Sales and Gray [1996]. In this approach residue calculus is used to transform the Green's function to a ratio of polynomials. A symbolic algebra program is used to generate polynomial coefficients. Newton's method is used to find the roots of the polynomial. Our implementation can handle multiple orthotropic materials; no precomputations are required.

- We have developed a fully 3D Galerkin BEM program. Coding for this program has been completed, we are currently in the testing and debugging phase. This code includes linear and quadratic order elements, and has been interfaced with our FRANC3D program that we use to generate models, meshes, and boundary conditions.

In summary, we have successfully completed the work tasks specified in our proposal. In addition, we have gone substantially beyond the original Phase I objectives towards creating a capability for 3D boundary element analysis of composite joints with discrete damage. We believe that we have demonstrated that the symmetric-Galerkin BEM formulation is well suited for this task, and that Fracture Analysis Consultants has the expertise to pursue this objective.

References

Bittencourt, T., Ingraffea, A. R., “Three-Dimensional Cohesive Crack Analysis of Short-Rod Specimens”, *Fracture Mechanics: 25th Volume,* ASTM-STP 1220, 46-60, 1995.

Blanford, G.E., Ingraffea, A.R., and Liggett J.A., "Two-dimensional stress intensity factor computations using the boundary element method," *Int J Num Meth Engng*, 17, 387-404, 1981.

Bonnet, M. and Bui, H.D., "Regularization of the displacement and traction BIE for 3D elastodynamics using indirect methods," in *Advances in Boundary Element Techiques*, Kane J.H. *et. al.* (eds), chapter 1, pp 1-29, Springer-Verlag, Berlin, 1993.

Bogdanovich, A. E., "Spline Function Aided Analysis of Inhomogeneous Materials and Structures", *Local Mechanics Concepts for Composite Material Systems,* J. N. Reddy and K. L. Reifsnyder (Eds.), Springer-Verlag, 355-382, 1992.

Boone, T. J., Wawrzynek. P.A., and Ingraffea, A. R., "Finite Element Modeling of Fracture Propagation in Orthotropic Materials," *Engineering Fracture Mechanics*, 26, 2, 185 - 201, 1987.

Brebbia, C.A., and Dominguez, J., *Boundary Elements, An Introductory Coarse*, McGraw-Hill, New York, 1989.

Chang, C., and Mear, M.E., "A boundary element method for two dimensional linear elastic fracture analysis," *Int J Fracture*, 74, pp. 219-251, 1995.

Crouch, S.L. and Selcuk, S., "Two-dimensional direct boundary integral method for multilayered elastic media," *Int J Rock Mech and Mining Sciences & GeoMech Abs*, 29 (5), pp 491-501, 1992.

Crouch, S.L. and Starfield, A.M., *Boundary Element Methods in Solid Mechanics*, George Allen & Unwin., London, 1983

Cruse, T.A., "Numerical solutions in three-dimensional elastostatics," *Int J Solids Structures*, 5, 1259-1274, 1969.

Cruse, T.A., *Boundary Element Analysis in Computational Fracture Mechanics*, Kluwer Academic Publishers, 1988.

Chamis, C. C., "Mechanics of Composite Materials: Past, Present, and Future", *J. Composites Tech. & Res.,* 11, 1, 3-14, 1989.

Dederichs, P.H. and Liebfried, G., "Elastic Green's function for anisotropic cubic crystals," *Phys Rev*, 188, pp. 1175-1183, 1969.

Ghandi, K.R., "Analysis of an inclinded crack centrally placed in an orhotropic rectangular plate," *Journal of Strain Analysis*, 7, 157-162, 1972.

Gray, L.J. and Paulino, G.H., "Symmetric Galerkin boundary integral fracture analysis for plane orthotropic elasticity," Proceeding, IUTAM Symposium on Innovative Computational Methods for Fracture and Damage, Dublin, Ireland, July 1996.

Gray, L.J., Balakrishna, C., and Kane, J.H., "Symmetric Galerkin boundary integral fracture analysis," *Engineering Analysis with Boundary Elements*, 15, 103-109, 1995.

Gray, L. J., "Symbolic Computation of Hypersingular Boundary Integrals," in *Advances in Boundary Element Methods*, J. H. Kane *et al.*, (Eds), 157-172, Springer-Verlag, 1993.

Gray, L. J., Martha, L. F., Ingraffea, A. R., "Hypersingular Integrals in Boundary Element Fracture Analysis," *Int J Num Meth Engr,* 29, 1135-1158, 1990.

Gray, L. J. , "Boundary Element Method for Regions with Thin Internal Cavities," *Engng. Analy.,* 6, 180-184, 1989.

Guimarães, S. and Telles, J.C.F., "On the hypersingular boundary-element formulation for fracture-mechanics applications," *Engng Analysis with Boundary Elements*, 13 (4), pp 353-363, 1994.

Hong, H.K. and Chen, J.T., "Derivation of integral equations in elasticity," *J Engng Mech* (ASCE), 114 (6), pp 1028-1044, 1988.

Joh, D., An experimental study of frictional phenomena around the pin joints of plates using Moire inferometry. Ph.D. Dissertation, Dept. Engineering Science and Mechanics, Virginia Polytechnic Institute and State University, Blacksburg, VA, 1986.

Krishnasami, G., Rizzo, F.J., and Rudolphi, T.J., "Hypersingular boundary integral equations: Their occurrence, interpretation, regularization, a computation," In *Developments in Boundary Element Methods*, Banerjee, P.K. and Kobayashi, S. (eds). volume 7, chapter 6, pp 207-252, Elsevier, London, 1992.

Lutz, E., Gray, L., Ingraffea, A. R., "An Overview of Integration Methods for Hypersingular Boundary Integrals," *Boundary Elements XIII*, C. Brebbia and G. Gipson, (Eds.), Computational Mechanics Publications, Elsevier Applied Science, 913-925, 1991.

Martha, Luiz F., Gray, L.J., Ingraffea, A. R., "Three-Dimensional Fracture Simulation with a Single-Domain, Direct Boundary Element Formulation," *Int J Num Meth Engr*, 35, 1992.

Maier, P.A., and Polizzotto, C., "A Galerkin approach to boundary element elasto-plastic analysis," Computer Meth in Appl Mech Engng, 60, 175-194, 1987

Pagano, N. J., "Stress Fields in Composite Laminates", *Int. J. Solids Structures*, 14, 385-400, 1978.

Paulino, G.H., Novel formulations of the boundary element method for fracture mechanics and error estimation, Ph.D. dissertation, Cornell University, Ithaca, New York, U.S.A.

Rizzo, F.J. and Shippy, D.J., "A formulation and solution for the general non-homogenous elastic inclusion problem," *Int J Solids Structures*, 4, 1161-1173, 1968.

Rao, A.K., "Elastic analysis of pin joints,", *Computers & Structures*, 9, 125-144, 1978.

Sales, M., Gray, L. J., "Evaluation of the Anisotropic Green's Function and its Derivatives," *J. Computers Structures*, Special Issue: Symposium on Industrial Applications and Recent Developments in the Boundary Element Method, to appear, 1996.

Schclar, N.A., *Anisotproic analysis using boundary element,* Computational Mechanics Publications, Southampton and Boston, 1994.

Sousa, J., Carter, B., Ingraffea, A. R., "Numerical Simulation of 3D Hydraulic Fracture Using Newtonian and Power-Law Fluids", *Int J Rock Mech Min Sci. & Geomech Abstr*, 30, 7, 1265-1271, 1993.

Sirtori, S., Maier, G., Novati, G., and Miccoli, S., "A Galerkin symmetric boundary-element method in elasticity: formulation and implementation," *Int J Num Meth Engng*, 35, pp 255-282, 1992.

Wang, C.Y., "Green's tensors for solid of general anisotropy," in *Boundary Elements XVIII*, C.A., Brebbia, ed. Southampton, Computational Mechancis, 1995.

Wang, A. S. D., Crossman, F. W., "Edge Effects on Thermally Induced Stresses in Composite Laminates," *J. Composite Materials*, 11, 300-312, 1977.

Wawrzynek, P.A., Discrete modeling of crack propagation: theoretical aspects and implementation issues in two and three dimensions, Ph.D. Dissertation, Cornell University, Ithaca, New York, 1991

Wilson, R.B., and Cruse, T.A., "Efficient implementation of anisotropic three dimensional boundary-integral equation stress analysis," *Int J. Numer Meth Engrg*, 12 (1978), pp. 1383-1397.

Yogeswaren, E.K. and Reddy, J.N. (1988) A study of contact stresses in pin-loaded orthotropic plates, *Comp &Struct*, Vol 30, pp 1067-1077.

Appendix A - Details of the 3D SG-BEM Formulation

Symbols:

a_j	- coefficients of linear term of Taylor expansion of distance between p and q, (j = 1..3)
a	- RMS norm of the a vector
C	- material constitutive matrix (6 x 6)
$dU_{kk\,jj,ll}$	- matrix of derivatives of the traction kernel (kk = 1..3, jj = 1..3, ll = 1..3)
$d\hat{U}_{kk\,jj,ll}$	- matrix of derivatives of the traction kernel for a truncated Taylor expanion in polar coordinates (kk = 1..3, jj = 1..3, ll = 1..3)
E_P, E_Q	- elements for integration
$\tilde{G}_{kk\,jj}$	- normalized Green's function (kk = 1..3, jj = 1..3)
$\tilde{G}_{kk\,jj}$	- truncated Taylor expansion of $\tilde{G}_{kk\,jj}$
gp	- Gauss point in P element
gq	- Gauss point in Q element
$\|J\|$	- jacobian determinant
$\|\hat{J}\|$	- truncated Taylor expansion of $\|J\|$ in polar coordinates
L	- distance from gauss point to element edge for element singular polar integrations
P	- integration source point
p_{jj}	- integration source point Cartesian coordinates (jj = 1..3)
Q	- integration field point
q_{jj}	- integration field point Cartesian coordinates (jj = 1..3)
$\hat{q}_{jj}$	- field point coordintates for a truncated Taylor expansion of r in polar coordinates (jj = 1..3)
r	- distance between source and destination point
$\hat{r}$	- truncated Taylor expansion of r in polar coordinates
$T_{kk\,jj}$	- displacement kernel (kk = 1..3, jj = 1..3)
$\hat{T}_{kk\,jj}$	- truncated Taylor expansion of $T_{kk\,jj}$ in polar coordinates
$U_{kk\,jj}$	- traction kernel (kk = 1..3, jj = 1..3)
u_{kk}	- displacement vector (kk = 1..3)
w	- Gauss integration weight

β_i - polynomial coefficients for a polar expansion of the shape functions (i = 0..3)

$\bar{\varepsilon}_{jj}$ - vector of strain vectors (jj = 1..3)

$\hat{\bar{\varepsilon}}_{jj}$ - vector of strain vectors for truncated Taylor expansion in polar coordinates (jj = 1..3)

γ_i - polynomial coefficients for a polar expansion of r (i = 0..4)

$\bar{\sigma}_{jj}$ - vector of stress vectors (jj = 1..3)

$\hat{\bar{\sigma}}_{jj}$ - vector of stress vectors for truncated Taylor expansion in polar coordinates (jj = 1..3)

τ_{kk} - traction vector (kk = 1..3)

ψ_k - shape function (k = 1..number of element nodes)

θ - angle for polar and sherical coordinates

ϕ - angle for spherical coordinates

ρ - distance for polar coordinates

η, ξ - p element parametric coordinates

$\bar{\eta}, \bar{\xi}$ - q element parametric coordinates

Governing Equations:

The governing boundary element equation is

$$u_{kk}(P) + \int T_{kk\,jj}(P,Q)\, u_{jj}(Q)\, dQ = \int U_{kk\,jj} \tau_{jj}(Q) dQ \; . \tag{1}$$

In terms of a Galerkin formulation, this can be rewritten as

$$\int \psi_k(P) u_{kk}(P) + \int \psi_k(P) \int T_{kk\,jj}(P,Q)\, u_{jj}(Q)\, dQ = \int \psi_k(P) \int U_{kk\,jj} \tau_{jj}(Q) dQ \; , \tag{2}$$

with the traction kernel function defined, in terms of spherical coordinates, as

$$U_{kk\,jj} = \frac{1}{8\pi^2 r} \tilde{G}_{kk\,jj}(\theta,\phi). \tag{3}$$

The displacement kernel function, T, is derived from the derivative of U with respect to the three coordinate directions,

$$dU_{kk\,jj,ll} = \frac{1}{8\pi^2} \left[-\frac{q_{ll} - p_{ll}}{r^3} \tilde{G}_{kk\,jj} + \frac{1}{r} \frac{\partial \tilde{G}_{kk\,jj}}{\partial \theta} \frac{\partial \theta}{\partial q_{ll}} + \frac{1}{r} \frac{\partial \tilde{G}_{kk\,jj}}{\partial \phi} \frac{\partial \phi}{\partial q_{ll}} \right] \tag{4}$$

from which can be formulated a strain vector

$$\bar{\varepsilon}_{jj} = \begin{Bmatrix} \bar{\varepsilon}_x \\ \bar{\varepsilon}_y \\ \bar{\varepsilon}_z \\ \bar{\varepsilon}_{xy} \\ \bar{\varepsilon}_{yz} \\ \bar{\varepsilon}_{zx} \end{Bmatrix}_{jj} = \begin{Bmatrix} dU_{1jj,1} \\ dU_{2jj,2} \\ dU_{3jj,3} \\ \frac{1}{2}\left(dU_{1jj,2} + dU_{2jj,1}\right) \\ \frac{1}{2}\left(dU_{2jj,3} + dU_{3jj,2}\right) \\ \frac{1}{2}\left(dU_{3jj,1} + dU_{1jj,3}\right) \end{Bmatrix}. \tag{5}$$

Using the constitutive relationship

$$\bar{\sigma}_{jj} = C_{6\times 6}\,\bar{\varepsilon}_{jj} \tag{6}$$

the displacement kernel can be defined,

$$T_{kk\,jj} = \begin{bmatrix} \bar{\sigma}_x & \bar{\sigma}_{xy} & \bar{\sigma}_{zx} \\ \bar{\sigma}_{xy} & \bar{\sigma}_y & \bar{\sigma}_{yz} \\ \bar{\sigma}_{zx} & \bar{\sigma}_{yz} & \bar{\sigma}_z \end{bmatrix}_{jj} \begin{Bmatrix} n_x \\ n_y \\ n_z \end{Bmatrix} \tag{7}$$

Spherical coordinates are usually defined as follows:

$$\left(\frac{q_1 - p_1}{r}, \frac{q_2 - p_2}{r}, \frac{q_3 - p_3}{r}\right) = (\cos\theta\sin\phi, \sin\theta\sin\phi, \cos\phi) \tag{8}$$

with $0 \le \phi \le \pi$, $0 \le \theta \le 2\pi$. In this case,

$$\frac{\partial\theta}{\partial q_1} = -\frac{q_2 - p_2}{r^2}\frac{1}{\sin^2\phi}, \quad \frac{\partial\theta}{\partial q_2} = \frac{q_1 - p_1}{r^2}\frac{1}{\sin^2\phi}, \quad \frac{\partial\theta}{\partial q_3} = 0, \text{ and} \tag{9}$$

$$\frac{\partial\phi}{\partial q_1} = \frac{(q_1 - p_1)}{r^2}\frac{\cos\phi}{\sin\phi}, \quad \frac{\partial\phi}{\partial q_2} = \frac{(q_2 - p_2)}{r^2}\frac{\cos\phi}{\sin\phi}, \quad \frac{\partial\phi}{\partial q_3} = -\left(\frac{q_1 - p_1}{r^2}\cos\theta + \frac{q_2 - p_2}{r^2}\sin\theta\right). \tag{10}$$

However, this gives rise to a singular point along the z axis. Therefore, in cases where $(q_3 - p_3)/r$ is > 0.7071, the following alternative spherical coordinate system defintion is used,

$$\left(\frac{q_1 - p_1}{r}, \frac{q_2 - p_2}{r}, \frac{q_3 - p_3}{r}\right) = (\cos\phi, \cos\theta\sin\phi, \sin\theta\sin\phi), \tag{11}$$

for which,

$$\frac{\partial\theta}{\partial q_1}=0,\quad \frac{\partial\theta}{\partial q_2}=-\frac{q_3-p_3}{r^2}\frac{1}{\sin^2\phi},\quad \frac{\partial\theta}{\partial q_3}=\frac{q_2-p_2}{r^2}\frac{1}{\sin^2\phi},\ \text{and} \tag{12}$$

$$\frac{\partial\phi}{\partial q_1}=-\left(\frac{q_2-p_2}{r^2}\cos\theta+\frac{q_3-p_3}{r^2}\sin\theta\right),\ \frac{\partial\phi}{\partial q_2}=\frac{q_2-p_2}{r^2}\frac{\cos\theta}{\sin\theta},\ \frac{\partial\phi}{\partial q_3}=\frac{q_3-p_3}{r^2}\frac{\cos\theta}{\sin\theta}. \tag{13}$$

Single integral term

The single integral term can be evaluated numerically as follows,

$$\int \psi_k(P)u_{kk}(P_j)\,dP = u_{kk}\sum_{gp}\psi_k(gp)\psi_j(gp)\left|J(gp)\right|w(gp). \tag{14}$$

Nonsingular integrals

Nonsingular integrals arise when elements P and Q have no nodes in common. In this case both integrals can be evaluated numerically; the traction integral being:

$$\begin{aligned}
&\int_{E_p}\psi_k(P)\int_{E_q}U_{kkjj}(P,Q)\,dP\\
&\quad=\int_{E_p}\psi_k\left|J_P\right|d\eta\,d\xi\int_{E_q}\psi_j U_{kkjj}\left|J_Q\right|d\bar{\eta}\,d\bar{\xi}\\
&\quad=\sum_{gp}\psi_k(gp)\left|J_P(gp)\right|w(gp)\sum_{gq}U_{kkjj}(gp,gq)\,\psi_j(gq)\left|J_Q(gq)\right|w(gq)\\
&\quad=\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)\left|J_P(gp)\right|w(gp)\sum_{gq}\frac{\tilde{G}_{kkjj}(gp,gq)}{r}\,\psi_j(gq)\left|J_Q(gq)\right|w(gq)
\end{aligned} \tag{15}$$

and the displacement integral is

$$\begin{aligned}
&\int_{E_P}\psi_k(P)\int_{E_Q}T_{kkjj}(P,Q)\,dQ\,dP\\
&\quad=\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)\left|J_P(gp)\right|w_P(gp)\sum_{gq}\tilde{T}_{kkjj}(gp,gq)\,\psi_j(gq)\,w(gq)
\end{aligned} \tag{16}$$

Note that in this case, the Q element jacobian determinant is included in the normal vector used to evaluate the displacement kernel, T Eq. (7).

Coincident Integrations

Coincident integrations occur when P and Q are the same element. In this case the integrations are singular, so they are decomposed into a nonsingular portion, solved numerically, and a singular portion, solved analytically. In order to find a form of the singular integral that can be evaluated analytically, the normal area integration is transformed to a polar integration centered at each of the elements gauss points.

To do this, define the Q gauss points as polar expansions about the P points, which are located at the element's gauss integration points,

$$\overline{\eta}=\eta+\rho\cos\theta, \text{ and } \overline{\xi}=\xi+\rho\sin\theta. \tag{17}$$

One can then define the vector from p to q in terms of ρ,

$$q_\alpha-p_\alpha=a_\alpha\rho+b_\alpha\rho^2+c_\alpha\rho^3, \quad \alpha=1,2,3 \tag{18}$$

$$\begin{aligned} r^2=&(a_1^2+a_2^2+a_3^2)\rho^2+2(a_1b_1+a_2b_2+a_3b_3)\rho^3+\\ &(2a_1c_1+b_1^2+2a_2c_2+b_2^2+2a_3c_3+b_3^2)\rho^4+\\ &2(b_1c_1+b_2c_2+b_3c_3)\rho^5+(c_1^2+c_2^2+c_3^2)\rho^6 \end{aligned} \tag{19}$$

or

$$r^2=\gamma_0\rho^2+\gamma_1\rho^3+\gamma_2\rho^4+\gamma_3\rho^5+\gamma_4\rho^6. \tag{20}$$

If we look at the first term in this expansion, $a=\sqrt{a_1^2+a_2^2+a_3^2}$, one can **define**

$$\hat{q}_\alpha=p_\alpha+a_\alpha\rho, \tag{21}$$

$$\hat{r}=a\rho, \tag{22}$$

$$\left|\hat{J}_Q\right|=\left|J_Q\right|_{\rho=0}=\left|J_P(\eta,\xi)\right|, \tag{23}$$

$$\hat{\tilde{G}}_{kkjj}=\tilde{G}\left(\frac{a_\alpha}{a}\right), \tag{24}$$

$$d\hat{U}_{kkjj,ll}=\frac{1}{8\pi^2}\left[-\frac{\hat{p}_{ll}-p_{ll}}{\hat{r}^3}\hat{\tilde{G}}_{kkjj}+\frac{1}{\hat{r}}\frac{\partial\hat{\tilde{G}}_{kkjj}}{\partial\theta}\frac{\partial\theta}{\partial q_{ll}}+\frac{1}{\hat{r}}\frac{\partial\hat{\tilde{G}}_{kkjj}}{\partial\phi}\frac{\partial\phi}{\partial q_{ll}}\right], \tag{25}$$

$$\hat{\varepsilon}_{jj}=\begin{Bmatrix} d\hat{U}_{1jj,1}\\ d\hat{U}_{2jj,2}\\ d\hat{U}_{3jj,3}\\ \frac{1}{2}\left(d\hat{U}_{1jj,2}+d\hat{U}_{2jj,1}\right)\\ \frac{1}{2}\left(d\hat{U}_{2jj,3}+d\hat{U}_{3jj,2}\right)\\ \frac{1}{2}\left(d\hat{U}_{3jj,1}+d\hat{U}_{1jj,3}\right) \end{Bmatrix}, \tag{26}$$

$\hat{\bar{\sigma}} = C\hat{\bar{\varepsilon}}$, and (27)

$\hat{\tilde{T}} = \hat{\bar{\sigma}}\hat{n}$. (28)

Then, the traction integral becomes,

$$
\begin{aligned}
&\frac{1}{8\pi^2}\int_{E_P}\psi_k(P)\,dP\int_{E_Q}\psi_j(Q)\frac{1}{r}\tilde{G}_{kkjj}(P,Q)\,dQ \\
&\quad= \frac{1}{8\pi^2}\int_E \psi_k |J_P|\,d\eta\, d\xi \int_E \psi_j \frac{1}{r}\tilde{G}_{kkjj}|J_Q|\,d\bar{\eta}\, d\bar{\xi} \\
&\quad= \frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\int_{E_Q}\psi_j\frac{1}{r}\tilde{G}_{kkjj}|J_Q|\,\rho\, d\rho\, d\theta \\
&\quad= \frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\int_{E_Q}\left[\psi_j\frac{1}{r}\tilde{G}_{kkjj}|J_Q| - \psi_j\frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}|\hat{J}_Q|\right]\rho\, d\rho\, d\theta \\
&\qquad + \frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\int_{E_Q}\psi_j\frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}|\hat{J}_Q|\rho\, d\rho\, d\theta
\end{aligned}
\tag{29}
$$

The first part is well behaved, and can be integrated numerically,

$$
\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\sum_{gq}\psi_j(gq)\,w(gq)\rho\left[\frac{1}{r}\tilde{G}_{kkjj}|J_Q| - \frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}|\hat{J}_Q|\right]
\tag{30}
$$

and the second part is solved with a hybrid numeric/analytical approach. Expanding the shape functions in terms of polar coordinates,

$$
\psi = \beta_0 + \beta_1(\theta)\rho + \beta_2(\theta)\rho^2 + \beta_3(\theta)\rho^3
\tag{31}
$$

then,

$$
\begin{aligned}
&\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\int_0^{2\pi}d\theta\int_0^{L(\theta)}\psi_j\frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}|\hat{J}_Q|\,\rho\, d\rho \\
&\quad= \frac{1}{8\pi^2}|\hat{J}_Q|\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\int_0^{2\pi}\frac{\hat{\tilde{G}}_{kkjj}}{a}d\theta\int_0^{L(\theta)}\psi_j\,\rho\, d\rho \\
&\quad= \frac{1}{8\pi^2}|\hat{J}_Q|\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\times \\
&\qquad\qquad \sum_{\Delta\theta}w(\Delta\theta)\frac{\hat{\tilde{G}}_{kkjj}}{a}\left[\beta_0 L(\theta) + \frac{\beta_1(\theta)L^2(\theta)}{2} + \frac{\beta_2(\theta)L^3(\theta)}{3} + \frac{\beta_3(\theta)L^4(\theta)}{4}\right]
\end{aligned}
\tag{32}
$$

The displacement part is:

$$\begin{aligned}
&\frac{1}{8\pi^2}\int_{E_P}\psi_k(P)dP\int_{E_Q}\psi_j(Q)T_{kkjj}(P,Q)\,dQ\\
&\quad=\frac{1}{8\pi^2}\int_E\psi_k|J_P|d\eta\,d\xi\int_E\psi_jT_{kkjj}|J_Q|\,d\bar{\eta}\,d\bar{\xi}\\
&\quad=\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\int_{E_Q}\psi_jT_{kkjj}|J_Q|\,\rho\,d\rho\,d\theta\\
&\quad=\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\int_{E_Q}\left[\psi_jT_{kkjj}|J_Q|-\psi_j\hat{T}_{kkjj}|\hat{J}_Q|\right]\rho\,d\rho\,d\theta\\
&\qquad+\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\int_{E_Q}\psi_j\hat{T}_{kkjj}|\hat{J}_Q|\rho\,d\rho\,d\theta
\end{aligned}\tag{33}$$

Again, the first part is well behaved, and can be integrated numerically,

$$\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\sum_{gq}\psi_j(gq)\,w(gq)\rho\left[T_{kkjj}|J_Q|-\hat{T}_{kkjj}|\hat{J}_Q|\right].\tag{34}$$

The second part is solved with a hybrid numeric/analytical approach,

$$\begin{aligned}
&\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\int_0^{2\pi}d\theta\int_0^{L(\theta)}\psi_j\hat{T}_{kkjj}|\hat{J}_Q|\,\rho\,d\rho\\
&\quad=\frac{1}{8\pi^2}|\hat{J}_Q|\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\int_0^{2\pi}d\theta\int_0^{L(\theta)}\hat{T}_{kkjj}\psi_j\,\rho\,d\rho\\
&\quad=\frac{1}{8\pi^2}|\hat{J}_Q|\sum_{gp}\psi_k(gp)|J_P(gp)|w(gp)\sum_{\Delta\theta}w(\Delta\theta)\int_0^{L(\theta)}\hat{T}_{kkjj}\psi_j\,\rho\,d\rho
\end{aligned}\tag{35}$$

where $\int_0^{L(\theta)}\hat{T}_{kkjj}\psi_j\,\rho\,d\rho$ is evaluated using symbolic computation.

Vertex Adjacent Integrations

When the two elements share one common node, the integration is said to be "vertex adjacent". For this case, the traction integral is:

$$\begin{aligned}
&\frac{1}{8\pi^2}\int_{E_P}\psi_k(P)dP\int_{E_Q}\psi_j(Q)\frac{1}{r}\tilde{G}_{kkjj}(P,Q)\,dQ\\
&\quad=\frac{1}{8\pi^2}\int_E\psi_k(P)dP\int_E\left[\psi_j\frac{1}{r}\tilde{G}_{kkjj}-\psi_j\frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}\right]dQ\\
&\qquad+\frac{1}{8\pi^2}\int_E\psi_k(P)dP\int_E\psi_j\frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}\,dQ
\end{aligned}\tag{36}$$

The first, well behaved, part is integrated numerically,

$$\begin{aligned}&\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)\left|J_p(gp)\right|w(gp)\sum_{gq}\psi_j(gq)\frac{1}{r}\tilde{G}_{kkjj}(gp,gq)\left|J_q(gq)\right|w(gq)\\&-\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)\left|\hat{J}_p(gp)\right|w(gp)\sum_{gq}\psi_j(gq)\frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}(gp,gq)\left|\hat{J}_q(gq)\right|w(gq)\end{aligned}\qquad(37)$$

The second part is evaluated with a hybrid numerical/analytical approach,

$$\begin{aligned}&\frac{1}{8\pi^2}\int_{\theta_{P1}}^{\theta_{P2}}\int_0^{L_P(\theta_P)}\psi_j(\rho_P,\theta_P)\rho_P d\rho_P d\theta_P\int_{\theta_{Q1}}^{\theta_{Q2}}\int_0^{L_Q(\theta_Q)}\psi_k(\rho_Q,\theta_Q)\frac{\hat{\tilde{G}}_{kkjj}}{\hat{r}}\rho_Q\,d\rho_Q\,d\theta_Q\\&=\frac{1}{8\pi^2}\int_{\theta_{P1}}^{\theta_{P2}}\int_{\theta_{Q1}}^{\theta_{Q2}}\int_0^{L_P(\theta_P)}\psi_j(\rho_P,\theta_P)\rho_P d\rho_P d\theta_Q d\theta_P\int_0^{L_Q(\theta_Q)}\psi_k(\rho_Q,\theta_Q)\frac{\hat{\tilde{G}}_{kkjj}}{\hat{r}}\rho_Q\,d\rho_Q\\&=\frac{1}{8\pi^2}\sum_{g\theta_P}w(g\theta_P)\left|\hat{J}_P(g\theta_P)\right|\sum_{g\theta_Q}w(g\theta_Q)\left|\hat{J}_Q(g\theta_Q)\right|\sum_{g\rho_P}w(g\rho_P)\psi_j(g\rho_P,g\theta_P)\rho_P\times\\&\int_0^{L_Q(\theta_Q)}\psi_k(\rho_Q,\theta_Q)\frac{\hat{\tilde{G}}_{kkjj}}{\hat{r}}\rho_Q\,d\rho_Q\end{aligned}\qquad(38)$$

with the analytical expressions for $\int_0^{L_Q(\theta_Q)}\psi_k(\rho_Q,\theta_Q)\frac{\hat{\tilde{G}}_{kkjj}}{\hat{r}}\rho_Q\,d\rho_Q$ being determined using symbolic computations, the resulting expressions are too long to be repeated here.

For the displacement integral,

$$\begin{aligned}&\frac{1}{8\pi^2}\int_{E_P}\psi_k(P)dP\int_{E_Q}\psi_j(Q)T_{kkjj}(P,Q)\,dQ\\&=\frac{1}{8\pi^2}\int_E\psi_k(P)dP\int_E\left[\psi_jT_{kkjj}-\psi_j\hat{T}_{kkjj}\right]dQ+\frac{1}{8\pi^2}\int_E\psi_k(P)dP\int_E\psi_j\hat{T}_{kkjj}\,dQ\end{aligned}\qquad(39)$$

the first part is integrated numerically,

$$\begin{aligned}&\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)\left|J_p(gp)\right|w(gp)\sum_{gq}\psi_j(gq)T_{kkjj}(gp,gq)\left|J_q(gq)\right|w(gq)\\&-\frac{1}{8\pi^2}\sum_{gp}\psi_k(gp)\left|\hat{J}_p(gp)\right|w(gp)\sum_{gq}\psi_j(gq)\hat{T}_{kkjj}(gp,gq)\left|\hat{J}_q(gq)\right|w(gq)\end{aligned}\qquad(40)$$

The second part is:

$$
\begin{aligned}
&\frac{1}{8\pi^2}\int_{\theta_{P1}}^{\theta_{P2}}\int_0^{L_P(\theta_P)}\psi_j(\rho_P,\theta_P)\rho_P d\rho_P d\theta_P\int_{\theta_{Q1}}^{\theta_{Q2}}\int_0^{L_Q(\theta_Q)}\psi_k(\rho_Q,\theta_Q)\hat{T}_{kkjj}\,\rho_Q\,d\rho_Q\,d\theta_Q \\
&\quad=\frac{1}{8\pi^2}\int_{\theta_{P1}}^{\theta_{P2}}\int_{\theta_{Q1}}^{\theta_{Q2}}\int_0^{L_P(\theta_P)}\psi_j(\rho_P,\theta_P)\rho_P d\rho_P d\theta_Q d\theta_P\int_0^{L_Q(\theta_Q)}\psi_k(\rho_Q,\theta_Q)\hat{T}_{kkjj}\,\rho_Q\,d\rho_Q \\
&\quad=\frac{1}{8\pi^2}\sum_{g\theta_P}w(g\theta_P)\left|\hat{J}_P(g\theta_P)\right|\sum_{g\theta_Q}w(g\theta_Q)\left|\hat{J}_Q(g\theta_Q)\right|\sum_{g\rho_P}w(g\rho_P)\psi_j(g\rho_P,g\theta_P)\rho_P\times \\
&\quad\int_0^{L_Q(\theta_Q)}\psi_k(\rho_Q,\theta_Q)\hat{T}_{kkjj}\,\rho_Q\,d\rho_Q
\end{aligned} \tag{41}
$$

with the analytical expressions for $\int_0^{L_Q(\theta_Q)}\psi_k(\rho_Q,\theta_Q)\hat{T}_{kkjj}\,\rho_Q\,d\rho_Q$ being determined using symbolic computations.

Edge Adjacent Integrations

The traction integral for this case is the same as Eq. (x1). Again the first part is evaluated numericaly. In this case, however, integration over the *u* and *v* coordinates in the *P* element are done independently,

$$
\begin{aligned}
&\frac{1}{8\pi^2}\int_u\int_v\psi_k(u,v)\int_{E_Q}\left[\psi_j(Q)\frac{1}{r}\tilde{G}_{kkjj}(P,Q)-\hat{\psi}_j(Q)\frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}(P,Q)\right]dQ \\
&\quad=\frac{1}{8\pi^2}\sum_{gu}w(gu)\sum_{gv}\psi_k(gu,gv)\left|J_P(gu,gv)\right|w(gv)\sum_{gq}\psi_j(gq)\frac{1}{r}\tilde{G}_{kkjj}\left|J_Q(gq)\right|w(gq) \\
&\quad-\frac{1}{8\pi^2}\sum_{gu}w(gu)\sum_{gv}\psi_k(gu,gv)\left|\hat{J}_P(gu,gv)\right|w(gv)\sum_{gq}\hat{\psi}_j(gq)\frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}\left|\hat{J}_Q(gq)\right|w(gq)
\end{aligned} \tag{42}
$$

The analytical part in this case is

$$
\begin{aligned}
&\frac{1}{8\pi^2}\int_u\int_v\psi_j(u,v)\,dv\,du\int_{\theta_1}^{\theta_2}\int_0^{L(\theta)}\psi_k(\rho,\theta)\frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}\,\rho\,d\rho\,d\theta \\
&\quad=\frac{1}{8\pi^2}\int_u\int_{\theta_1}^{\theta_2}\int_v\int_0^{L(\theta)}\psi_j(u,v)\psi_k(\rho,\theta)\frac{1}{\hat{r}}\hat{\tilde{G}}_{kkjj}\rho\,d\rho\,dv\,d\theta\,dv \\
&\quad=\frac{1}{8\pi^2}\sum_{gu}w(gu)\left|\hat{J}_P(gu)\right|\sum_{g\theta}w(g\theta)\left|\hat{J}_Q(g\theta)\right|\times \\
&\quad\sum_{gv}w(gv)\hat{\tilde{G}}_{kkjj}\int_0^{L(\theta)}\psi_j(gu,gv)\psi_k(\rho,g\theta)\frac{\rho}{\hat{r}}d\rho
\end{aligned} \tag{43}
$$

where $\int_0^{L(\theta)}\psi_j(gu,gv)\psi_k(\rho,g\theta)\frac{\rho}{\hat{r}}d\rho$ is integrated symbolically.

The displacement part is the same as Eq. (x3). the numerical part being,

$$
\begin{aligned}
&\frac{1}{8\pi^2}\int_u\int_v \psi_k(u,v)\int_{E_Q}\left[\psi_j(Q)\frac{1}{r}T_{kkjj}(P,Q)-\hat{\psi}_j(Q)\hat{T}_{kkjj}(P,Q)\right]dQ\\
&\quad=\frac{1}{8\pi^2}\sum_{gu}w(gu)\sum_{gv}\psi_k(gu,gv)\left|J_P(gu,gv)\right|w(gv)\sum_{gq}\psi_j(gq)T_{kkjj}\left|J_Q(gq)\right|w(gq)\\
&\quad-\frac{1}{8\pi^2}\sum_{gu}w(gu)\sum_{gv}\psi_k(gu,gv)\left|\hat{J}_P(gu,gv)\right|w(gv)\sum_{gq}\hat{\psi}_j(gq)\hat{T}_{kkjj}\left|\hat{J}_Q(gq)\right|w(gq)
\end{aligned}
\tag{44}
$$

The analytical part is

$$
\begin{aligned}
&\frac{1}{8\pi^2}\int_u\int_v \psi_j(u,v)\,dv\,du\int_{\theta_1}^{\theta_2}\int_0^{L(\theta)}\psi_k(\rho,\theta)\hat{T}_{kkjj}\,\rho\,d\rho\,d\theta\\
&\quad=\frac{1}{8\pi^2}\int_u\int_{\theta_1}^{\theta_2}\int_v\int_0^{L(\theta)}\psi_j(u,v)\psi_k(\rho,\theta)\hat{T}_{kkjj}\rho\,d\rho\,dv\,d\theta\,dv\\
&\quad=\frac{1}{8\pi^2}\sum_{gu}w(gu)\left|\hat{J}_P(gu)\right|\sum_{g\theta}w(g\theta)\left|\hat{J}_Q(g\theta)\right|\times\\
&\quad\sum_{gv}w(gv)\int_0^{L(\theta)}\hat{T}_{kkjj}\psi_j(gu,gv)\psi_k(\rho,g\theta)\,\rho\,d\rho
\end{aligned}
\tag{45}
$$

with $\int_0^{L(\theta)}\hat{T}_{kkjj}\psi_j(gu,gv)\psi_k(\rho,g\theta)\,\rho\,d\rho$ integrated using symbolic computations.